Red Hat Enterprise Linux 5
Installation Guide

Installing Red Hat Enterprise Linux 5 for all architectures

Red Hat Engineering Content Services
Rüdiger Landmann
Jack Reed
Petr Bokoč
David Cantrell
Hans De Goede
Jon Masters

Red Hat Enterprise Linux 5 Installation Guide

Installing Red Hat Enterprise Linux 5 for all architectures

Rüdiger Landmann
Red Hat Engineering Content Services
r.landmann@redhat.com

Jack Reed
Red Hat Engineering Content Services
jreed@redhat.com

Petr Bokoč
Red Hat Engineering Content Services
pbokoc@redhat.com

David Cantrell
dcantrell@redhat.com
VNC installation

Hans De Goede
hdgoede@redhat.com
iSCSI

Jon Masters
jcm@redhat.com
Driver updates

Red Hat Engineering Content Services

Edited by

Rüdiger Landmann
r.landmann@redhat.com

Jack Reed
jreed@redhat.com

Petr Bokoč
pbokoc@redhat.com

Legal Notice

Abstract

This Installation Guide documents relevant information regarding the installation of Red Hat Enterprise Linux 5.

Table of Contents

Part I. x86, AMD64, Intel® 64 and Itanium - Installation and Booting

The *Red Hat Enterprise Linux Installation Guide* for Intel and AMD 32-bit and 64-bit systems discusses the installation of Red Hat Enterprise Linux and some basic post-installation troubleshooting. Advanced installation options are covered in the second part of this manual.

Chapter 1. Itanium System Specific Information

1.1. Itanium System Installation Overview

Installing Red Hat Enterprise Linux on an Itanium system is different from installing Red Hat Enterprise Linux on an x86-based system. In general, the sequence of steps to a successful installation are the following:

1. Boot into the Extensible Firmware Interface (EFI) Shell.
2. If you cannot boot from the CD-ROM, make an LS-120 diskette from the boot image file provided with Red Hat Enterprise Linux.
3. Using the EFI Shell and the ELILO boot loader, load and run the kernel, and boot into the Red Hat Enterprise Linux installation program.

1.2. Itanium Systems — The EFI Shell

Before you start to install Red Hat Enterprise Linux on an Itanium, you must have a basic understanding of the EFI Shell, what it does, and the information it can provide.

The EFI Shell is a console interface used to launch applications (such as the Red Hat Enterprise Linux installation program), load EFI protocols and device drivers, and execute simple scripts. It is similar to a DOS console and can only access media that is FAT16 (VFAT) formatted.

The EFI Shell also contains common utilities that can be used on the EFI system partition. These utilities include **edit**, **type**, **cp**, **rm**, and **mkdir**. For a list of utilities and other commands, type **help** at the EFI Shell prompt.

The EFI Shell contains a boot loader called ELILO. Additional information on EFI can be found at the following URL:

```
http://developer.intel.com/technology/efi/index.htm
```

1.2.1. Itanium Systems — EFI Device Names

The **map** command can be used to list all devices and file systems that EFI can recognize. When your Itanium system boots into the EFI shell, it probes your system in the following order:

1. LS-120 drive (if it contains media)
2. IDE hard drives on the primary IDE interface
3. IDE hard drives on the secondary IDE interface
4. SCSI hard drives on the SCSI interface
5. CD-ROM drives on the IDE interface
6. CD-ROM drives on the SCSI interface

To view the results of this system poll, type the following command at the EFI Shell prompt:

```
map
```

The output is listed in the order the system was probed. So, all FAT16 file systems are listed first, then IDE hard drives, then SCSI hard drives, then IDE CD-ROM drives, and finally SCSI CD-ROM drives.

For example, output of the **map** command might look like the following:

```
Device mapping table
 fs0  : VenHw(Unknown Device:00)/HD(Part1,Sig00000000)
 fs1  : VenHw(Unknown Device:80)/HD(Part1,Sig00000000)
 fs2  : VenHw(Unknown Device:FF)/CDROM(Entry1)/HD(Part1,Sig00000000)
 blk0 : VenHw(Unknown Device:00)
 blk1 : VenHw(Unknown Device:00)/HD(Part1,Sig00000000)
 blk2 : VenHw(Unknown Device:80)
 blk3 : VenHw(Unknown Device:80)/HD(Part1,Sig00000000)
 blk4 : VenHw(Unknown Device:80)/HD(Part2,Sig00000000)
 blk5 : VenHw(Unknown Device:80)/HD(Part3,Sig00000000)
 blk6 : VenHw(Unknown Device:80)/HD(Part3,Sig00000000)/HD(Part1,Sig725F7772)
 blk7 : VenHw(Unknown Device:FF)
 blk8 : VenHw(Unknown Device:FF)/CDROM(Entry1)
 blk9 : VenHw(Unknown Device:FF)/CDROM(Entry1)/HD(Part1,Sig00000000)
```

In this example, there is an LS-120 diskette in the LS-120 drive as well as a CD-ROM in the CD-ROM drive. All the listings beginning with **fs** are FAT16 file systems that EFI can read. All the listings beginning with **blk** are block devices that EFI recognizes. Both the file systems and block devices are listed in the order they are probed. Therefore, **fs0** is the system partition on the LS-120, **fs1** is the system partition on the hard drive, and **fs2** is the system partition on the CD-ROM.

1.2.2. Itanium Systems — EFI System Partition

When partitioning your hard drive for Linux, you must create a system partition that is FAT16 (VFAT) formatted and has a mount point of **/boot/efi/**. This partition contains the installed Linux kernel(s) as well as the ELILO configuration file (**elilo.conf**). The **elilo.conf** file contains a list of kernels from which you can boot your system.

Chapter 2. Steps to Get You Started

2.1. Upgrade or Install?

For information to help you determine whether to perform an upgrade or an installation refer to Chapter 24, *Upgrading Your Current System*.

2.2. Is Your Hardware Compatible?

Hardware compatibility is particularly important if you have an older system or a system that you built yourself. Red Hat Enterprise Linux 5 should be compatible with most hardware in systems that were factory built within the last two years. However, hardware specifications change almost daily, so it is difficult to guarantee that your hardware is 100% compatible.

The most recent list of supported hardware can be found at:

```
http://hardware.redhat.com/hcl/
```

2.3. Do You Have Enough Disk Space?

Nearly every modern-day operating system (OS) uses *disk partitions*, and Red Hat Enterprise Linux is no exception. When you install Red Hat Enterprise Linux, you may have to work with disk partitions. If you have not worked with disk partitions before (or need a quick review of the basic concepts), refer to Chapter 26, *An Introduction to Disk Partitions* before proceeding.

The disk space used by Red Hat Enterprise Linux must be separate from the disk space used by other OSes you may have installed on your system, such as Windows, OS/2, or even a different version of Linux. For x86, AMD64, and Intel® 64 systems, at least two partitions (**/** and **swap**) must be dedicated to Red Hat Enterprise Linux. For Itanium systems, at least three partitions (**/**, **/boot/efi/**, and **swap**) must be dedicated to Red Hat Enterprise Linux.

Before you start the installation process, you must

- have enough *unpartitioned* [1] disk space for the installation of Red Hat Enterprise Linux, or
- have one or more partitions that may be deleted, thereby freeing up enough disk space to install Red Hat Enterprise Linux.

To gain a better sense of how much space you really need, refer to the recommended partitioning sizes discussed in Section 4.19.4, “Recommended Partitioning Scheme”.

If you are not sure that you meet these conditions, or if you want to know how to create free disk space for your Red Hat Enterprise Linux installation, refer to Chapter 26, *An Introduction to Disk Partitions*.

2.4. Can You Install Using the CD-ROM or DVD?

There are several methods that can be used to install Red Hat Enterprise Linux.

Installing from a CD-ROM or DVD requires that you have purchased a Red Hat Enterprise Linux product, you have a Red Hat Enterprise Linux 5.11 CD-ROM or DVD, and you have a DVD/CD-ROM drive on a system that supports booting from it.

If you do not already have a Red Hat Enterprise Linux 5.11 CD set or DVD, you can download one as an *ISO image file* from the Red Hat Customer Portal. Visit https://access.redhat.com/home and enter your Login and Password. Click on the **Downloads** link to obtain a list of all currently supported Red Hat Enterprise Linux products. If you do not already have a Login and Password for the Red Hat Customer Portal, visit https://access.redhat.com/downloads/ to purchase a subscription or obtain a free evaluation subscription. When you have obtained the image file, you can burn it to disk with the following procedure:

The exact series of steps that produces a CD from an image file varies greatly from computer to computer, depending on the operating system and disc burning software installed. Use this procedure as a general guide. You might be able to omit certain steps on your computer, or might have to perform some of the steps in a different order from the order described here.

Make sure that your disc burning software is capable of burning discs from image files. Although this is true of most disc burning software, exceptions exist.

In particular, note that the CD burning feature built into Windows XP and Windows Vista cannot burn CDs from images and that earlier Windows operating systems did not have any CD burning capability installed by default. Therefore, if your computer has a Windows operating system installed on it, you need a separate piece of software for this task. Examples of popular CD burning software for Windows that you might already have on your computer include **Nero Burning ROM** and **Roxio Creator**. If you use a Windows operating system on your computer and do not have disc burning software installed (or you are not sure that the software can burn discs from image files) **InfraRecorder** is a suitable alternative available from http://www.infrarecorder.org/, and is free and open-source.

The **Disk Utility** software installed by default with Mac OS X on Apple computers has the capability to burn CDs from images built into it already. Most widely-used CD burning software for Linux, such as **Brasero** and **K3b**, also includes this capability.

1. Insert a blank, writeable CD into your computer's CD or DVD burner. On some computers, a window opens and displays various options when you insert the disc. If you see a window like this, look for an option to launch your chosen disc burning program. If you do not see an option like this, close the window and launch the program manually.

2. Launch your disc burning program. On some computers, you can do this by right-clicking (or control-clicking) on the image file and selecting a menu option with a label like **Copy image to CD**, or **Copy CD or DVD image**. Other computers might provide you with a menu option to launch your chosen disc burning program, either directly or with an option like **Open With**. If none of these options are available on your computer, launch the program from an icon on your desktop, in a menu of applications such as the **Start** menu on Windows operating systems, or in the Mac **Applications** folder.

3. In your disc burning program, select the option to burn a CD from an image file. For example, in **Nero Burning ROM**, this option is called **Burn Image** and is located on the **File** menu.

   Note that you can skip this step when using certain CD burning software; for example, **Disk Utility** on Mac OS X does not require it.

4. Browse to the disc image file that you downloaded previously and select it for burning.

5. Click the button that starts the burning process.

Your BIOS may need to be changed to allow booting from your DVD/CD-ROM drive. For more information about changing your BIOS, refer to Section 4.4.1, “Booting the Installation Program on x86, AMD64, and Intel® 64 Systems”.

2.4.1. Alternative Boot Methods

Boot DVD/CD-ROM

If you can boot using the DVD/CD-ROM drive, you can create your own CD-ROM to boot the installation program. This may be useful, for example, if you are performing an installation over a network or from a hard drive. Refer to Section 2.4.2, “Making an Installation Boot CD-ROM” for further instructions.

USB pen drive

If you cannot boot from the DVD/CD-ROM drive, but you can boot using a USB device, such as a USB pen drive, the following alternative boot method is available.

Your system firmware must support booting from a USB device in order for this boot method to work. Refer to the hardware vendor's documentation for details on specifying the device from which the system boots.

> **USB devices may not be named as expected**
>
> When configuring partitions and file systems during installation, ensure you verify the USB device's size, name, and type. The order in which names are assigned to USB-attached storage devices can vary because certain devices may take longer to initialize than others. Consequently, a device may receive a different name than you expect, such as **sdc** instead of **sda**.

1. Make a copy of the Red Hat Enterprise Linux 5 installation files available. Either:

   A. Insert the Red Hat Enterprise Linux 5 installation DVD or CD-ROM#1.

   B. Mount an image of the Red Hat Enterprise Linux 5 installation DVD or CD-ROM#1.

   C. Ensure that the installation files are available on a network location accessible by your system, for example, on an NFS share that it can access.

2. Attach a USB flash drive to your system. The following steps presume a system that runs Red Hat Enterprise Linux 5.

3. Run **dmesg** to identify the device name for the drive. If you run **dmesg** shortly after you attach the drive, the device name appears in the most recent lines of output. For example, the following **dmesg** output shows a flash drive that receives the device name **/dev/sdb**:

```
Initializing USB Mass Storage driver...
scsi2 : SCSI emulation for USB Mass Storage devices
usb-storage: device found at 5
usb-storage: waiting for device to settle before scanning
usbcore: registered new driver usb-storage
USB Mass Storage support registered.
  Vendor: USB 2.0   Model: Flash Disk       Rev: 5.00
  Type:   Direct-Access                     ANSI SCSI revision: 02
SCSI device sdb: 2043904 512-byte hdwr sectors (1046 MB)
sdb: Write Protect is off
sdb: Mode Sense: 0b 00 00 08
sdb: assuming drive cache: write through
SCSI device sdb: 2043904 512-byte hdwr sectors (1046 MB)
sdb: Write Protect is off
sdb: Mode Sense: 0b 00 00 08
sdb: assuming drive cache: write through
```

```
sdb: sdb1
sd 2:0:0:0: Attached scsi removable disk sdb
sd 2:0:0:0: Attached scsi generic sg1 type 0
usb-storage: device scan complete
```

4. Unmount any partitions on the flash drive that are currently mounted. It is likely that your system automatically mounted any available partitions when you attached the flash drive.

   a. Use the **mount** command to find any mounted partitions on the flash drive. For example, the following output shows a single partition on **/dev/sdb** is mounted, the partition named **/dev/sdb1**:

   ```
   $ mount
   /dev/mapper/VolGroup00-LogVol00 on / type ext3 (rw)
   proc on /proc type proc (rw)
   sysfs on /sys type sysfs (rw)
   devpts on /dev/pts type devpts (rw,gid=5,mode=620)
   tmpfs on /dev/shm type tmpfs
   (rw,rootcontext="system_u:object_r:tmpfs_t:s0")
   /dev/sda1 on /boot type ext3 (rw)
   none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
   sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)
   /dev/sdb1 on /media/BOOTUSB type vfat
   (rw,nosuid,nodev,uid=500,utf8,shortname=mixed,flush)
   ```

   b. Unmount partitions with the **umount** command. For example, to unmount **/dev/sdb1**, run:

   ```
   umount /dev/sdb1
   ```

   Run **umount** for each partition on the flash drive that is mounted.

5. Use **fdisk** to partition the flash drive to contain a single partition only, with the following parameters:

   a. numbered **1**.

   b. partition type is set to **b** (W95 FAT32).

   c. flagged as bootable.

6. Run **mkdosfs** to format the partition created in the previous step as FAT. For example:

   ```
   mkdosfs /dev/sdb1
   ```

7. Mount the partition. For example:

   ```
   mount /dev/sdb1 /mnt
   ```

8. Copy the contents of the **isolinux/** directory of the installation DVD or CD-ROM#1 onto the flash drive.

9. Rename the configuration file from **isolinux.cfg** to **syslinux.cfg**. For example, if the flash drive is mounted on **/mnt**, run:

   ```
   cd /mnt/; mv isolinux.cfg syslinux.cfg
   ```

a. If necessary, edit **syslinux.cfg** for your particular environment. For example, to configure the installation to use a kickstart file shared over NFS, specify:

```
linux ks=nfs::/ks.cfg
```

10. Copy the **images/pxeboot/initrd.img** file from the installation DVD or CD-ROM#1 onto the flash drive.

11. Unmount the flash drive. For example:

```
umount /dev/sdb1
```

12. Make the USB flash drive bootable. For example:

```
syslinux /dev/sdb1
```

13. Mount the flash drive again. For example:

```
mount /dev/sdb1 /mnt
```

14. Install the **GRUB** boot loader on the USB flash drive. For example:

```
grub-install --root-directory=/mnt /dev/sdb
```

15. Verify that the USB flash drive has a /boot/grub directory. If it does not, create the directory manually; for example:

```
mkdir -p /mnt/boot/grub
```

16. Create the **boot/grub/grub.conf** file on the flash drive as follows:

```
default=0
timeout=5
root (hd1,0)
title Red Hat Enterprise Linux installer
kernel /vmlinuz
initrd /initrd.img
```

17. Unmount the flash drive. For example:

```
umount /dev/sdb1
```

18. Detach the USB flash drive.

19. Attach the USB disk to the system on which you wish to install Red Hat Enterprise Linux.

20. Boot the target system from the USB flash drive.

2.4.2. Making an Installation Boot CD-ROM

isolinux (not available for Itanium systems) is used for booting the Red Hat Enterprise Linux installation CD. To create your own CD-ROM to boot the installation program, use the following instructions:

Copy the **isolinux/** directory from the Red Hat Enterprise Linux DVD or CD #1 into a temporary directory (referred to here as ***<path-to-workspace>***) using the following command:

```
cp -r <path-to-cd>/isolinux/ <path-to-workspace>
```

Change directories to the ***<path-to-workspace>*** directory you have created:

```
cd <path-to-workspace>
```

Make sure the files you have copied have appropriate permissions:

```
chmod u+w isolinux/*
```

Finally, issue the following command to create the ISO image file:

```
mkisofs -o file.iso -b isolinux.bin -c boot.cat -no-emul-boot \
-boot-load-size 4 -boot-info-table -R -J -v -T isolinux/
```

Note

The above command was split into two lines for printing purposes only. When you execute this command, be sure to type it as a single command, all on the same line.

Burn the resulting ISO image (named **file.iso** and located in ***<path-to-workspace>***) to a CD-ROM as you normally would.

2.5. Preparing for a Network Installation

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

The Red Hat Enterprise Linux installation media must be available for either a network installation (via NFS, FTP, or HTTP) or installation via local storage. Use the following steps if you are performing an NFS, FTP, or HTTP installation.

The NFS, FTP, or HTTP server to be used for installation over the network must be a separate machine which can provide the complete contents of the installation DVD-ROM or the installation CD-ROMs.

Note

The Red Hat Enterprise Linux installation program has the ability to test the integrity of the installation media. It works with the CD, DVD, hard drive ISO, and NFS ISO installation methods. Red Hat recommends that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned CDs). To use this test, type the following command at the **boot:** prompt (prepend with **elilo** for Itanium systems):

```
linux mediacheck
```

> **Note**
>
> In the following examples, the directory on the installation staging server that will contain the installation files will be specified as ***/location/of/disk/space***. The directory that will be made publicly available via FTP, NFS, or HTTP will be specified as */publicly/available/directory*. For example, ***/location/of/disk/space*** may be a directory you create called **/var/isos**. ***/publicly/available/directory*** might be **/var/www/html/rhel5**, for an HTTP install.

To copy the files from the installation DVD or CD-ROMs to a Linux machine which acts as an installation staging server, perform the following steps:

- Create an iso image from the installation disk(s) using the following command (for DVDs):

  dd if=/dev/*dvd* of=/*location/of/disk/space*/RHEL5.iso

  where *dvd* refers to your DVD drive device.

2.5.1. Preparing for FTP and HTTP installation

If you are setting up an installation tree for NFS, FTP, or HTTP installations, you must copy the **RELEASE-NOTES** files and all files from the **RedHat** directory on all operating systems ISO images. On Linux and UNIX systems, the following process will properly configure the target directory on your server (repeat for each CD-ROM/ISO image):

1. Insert CD-ROM or DVD-ROM.
2. **mount /media/cdrom**
3. If you are installing the Server variant, run **cp -a /media/cdrom/Server <*target-directory*>**

   If you are installing the Client variant, run **cp -a /media/cdrom/Client <*target-directory*>**
4. **cp /media/cdrom/RELEASE-NOTES* <*target-directory*>** (Installation CD 1 or DVD only)
5. **cp /media/cdrom/images <*target-directory*>** (Installation CD 1 or DVD only)
6. **umount /media/cdrom**

(Where **<*target-directory*>** represents the path to the directory to contain the installation tree.)

> **Note**
>
> Do *not* copy the Supplementary ISO image, or any of the layered product ISO images, as this will overwrite files necessary for Anaconda's proper operation.
>
> These ISO images must be installed *after* Red Hat Enterprise Linux has been installed.

Next make sure that the ***/publicly/available/directory*** directory is shared via FTP or HTTP, and verify client access. You can check to see whether the directory is accessible from the server itself, and then from another machine on the same subnet that you will be installing to.

2.5.2. Preparing for an NFS install

For NFS installation it is not necessary to mount the iso image. It is sufficient to make the iso image itself available via NFS. You can do this by moving the iso image or images to the NFS exported directory:

- For DVD:

  ```
  mv /location/of/disk/space/RHEL5.iso /publicly/available/directory/
  ```

- For CDROMs:

  ```
  mv /location/of/disk/space/disk*.iso /publicly/available/directory/
  ```

Ensure that the ***/publicly/available/directory*** directory is exported via NFS via an entry in **/etc/exports**.

To export to a specific system:

```
/publicly/available/directory client.ip.address
```

To export to all systems use an entry such as:

```
/publicly/available/directory *
```

Start the NFS daemon (on a Red Hat Enterprise Linux system, use **/sbin/service nfs start**). If NFS is already running, reload the configuration file (on a Red Hat Enterprise Linux system use **/sbin/service nfs reload**).

Be sure to test the NFS share following the directions in the Red Hat Enterprise Linux Deployment Guide.

2.6. Preparing for a Hard Drive Installation

> **Note**
>
> Hard drive installations only work from ext2, ext3, or FAT file systems. If you have a file system other than those listed here, such as reiserfs, you will not be able to perform a hard drive installation.

Hard drive installations require the use of the ISO (or DVD/CD-ROM) images. An ISO image is a file containing an exact copy of a DVD/CD-ROM image. After placing the required ISO images (the binary Red Hat Enterprise Linux DVD/CD-ROMs) in a directory, choose to install from the hard drive. You can then point the installation program at that directory to perform the installation.

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

To prepare your system for a hard drive installation, you must set the system up in one of the following ways:

- Using a set of CD-ROMs, or a DVD — Create ISO image files from each installation CD-ROM, or from the DVD. For each CD-ROM (once for the DVD), execute the following command on a Linux system:

```
dd if=/dev/cdrom of=/tmp/file-name.iso
```

- Using ISO images — transfer these images to the system to be installed.

  Verifying that ISO images are intact before you attempt an installation, helps to avoid problems. To verify the ISO images are intact prior to performing an installation, use an **md5sum** program (many **md5sum** programs are available for various operating systems). An **md5sum** program should be available on the same Linux machine as the ISO images.

Note

The Red Hat Enterprise Linux installation program has the ability to test the integrity of the installation media. It works with the CD / DVD, hard drive ISO, and NFS ISO installation methods. Red Hat recommends that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned CDs). To use this test, type the following command at the **boot:** prompt (prepend with **elilo** for Itanium systems):

```
linux mediacheck
```

Additionally, if a file called **updates.img** exists in the location from which you install, it is used for updates to **anaconda**, the installation program. Refer to the file **install-methods.txt** in the **anaconda** RPM package for detailed information on the various ways to install Red Hat Enterprise Linux, as well as how to apply the installation program updates.

---

[1] Unpartitioned disk space means that available disk space on the hard drive(s) you are installing to has not been divided into sections for data. When you partition a disk, each partition behaves like a separate disk drive.

Chapter 3. System Specifications List

The most recent list of supported hardware can be found at http://hardware.redhat.com/hcl/.

This system specifications list will help you keep a record of your current system settings and requirements. Enter the corresponding information about your system in the list provided below as a handy reference to help make your Red Hat Enterprise Linux installation goes smoothly.

- *hard drive(s)*: type, label, size; e.g. IDE hda=40 GB
- *partitions*: map of partitions and mount points; e.g. **`/dev/hda1=/home`**, **`/dev/hda2=/`** (fill this in once you know where they will reside)
- *memory*: amount of RAM installed on your system; e.g. 512 MB, 1 GB
- *CD-ROM*: interface type; e.g. SCSI, IDE (ATAPI)
- *SCSI adapter*: if present, make and model number; e.g. BusLogic SCSI Adapter, Adaptec 2940UW
- *network card*: if present, make and model number; e.g. Tulip, 3COM 3C590
- *mouse*: type, protocol, and number of buttons; e.g. generic 3 button PS/2 mouse, MouseMan 2 button serial mouse
- *monitor*: make, model, and manufacturer specifications; e.g. Optiquest Q53, ViewSonic G773
- *video card*: make, model number and size of VRAM; e.g. Creative Labs Graphics Blaster 3D, 8MB
- *sound card*: make, chipset and model number; e.g. S3 SonicVibes, Sound Blaster 32/64 AWE
- *IP, DHCP, and BOOTP addresses*
- *netmask*
- *gateway IP address*
- *one or more name server IP addresses (DNS)*
- *domain name*: the name given to your organization; e.g. **`example.com`**
- *hostname*: the name of your computer; your personal choice of names; e.g. **`cookie`**, **`southpark`**

If any of these networking requirements or terms are unfamiliar to you, contact your network administrator for assistance.

Chapter 4. Installing on Intel® and AMD Systems

This chapter explains how to perform a Red Hat Enterprise Linux installation from the DVD/CD-ROM, using the graphical, mouse-based installation program. The following topics are discussed:

- Becoming familiar with the installation program's user interface
- Starting the installation program
- Selecting an installation method
- Configuration steps during the installation (language, keyboard, mouse, partitioning, etc.)
- Finishing the installation

4.1. The Graphical Installation Program User Interface

If you have used a *graphical user interface (GUI)* before, you are already familiar with this process; use your mouse to navigate the screens, click buttons, or enter text fields.

You can also navigate through the installation using the keyboard. The **Tab** key allows you to move around the screen, the Up and Down arrow keys to scroll through lists, **+** and **-** keys expand and collapse lists, while **Space** and **Enter** selects or removes from selection a highlighted item. You can also use the **Alt**+***X*** key command combination as a way of clicking on buttons or making other screen selections, where ***X*** is replaced with any underlined letter appearing within that screen.

Note

If you are using an x86, AMD64, or Intel® 64 system, and you do not wish to use the GUI installation program, the text mode installation program is also available. To start the text mode installation program, use the following command at the **boot:** prompt:

```
linux text
```

Refer to Section 4.3, “The Text Mode Installation Program User Interface” for a brief overview of text mode installation instructions.

It is highly recommended that installs be performed using the GUI installation program. The GUI installation program offers the full functionality of the Red Hat Enterprise Linux installation program, including LVM configuration which is not available during a text mode installation.

Users who must use the text mode installation program can follow the GUI installation instructions and obtain all needed information.

> **Note**
>
> If you are using an Itanium system, and you do not wish to use the GUI installation program, the text mode installation program is also available. To start the text mode installation program, type the following command at the EFI Shell prompt:
>
> ```
> elilo linux text
> ```

4.1.1. A Note about Virtual Consoles

The Red Hat Enterprise Linux installation program offers more than the dialog boxes of the installation process. Several kinds of diagnostic messages are available to you, as well as a way to enter commands from a shell prompt. The installation program displays these messages on five *virtual consoles*, among which you can switch using a single keystroke combination.

A virtual console is a shell prompt in a non-graphical environment, accessed from the physical machine, not remotely. Multiple virtual consoles can be accessed simultaneously.

These virtual consoles can be helpful if you encounter a problem while installing Red Hat Enterprise Linux. Messages displayed on the installation or system consoles can help pinpoint a problem. Refer to Table 4.1, “Console, Keystrokes, and Contents” for a listing of the virtual consoles, keystrokes used to switch to them, and their contents.

Generally, there is no reason to leave the default console (virtual console #6) for graphical installations unless you are attempting to diagnose installation problems.

Table 4.1. Console, Keystrokes, and Contents

console	keystrokes	contents
1	**ctrl**+**alt**+**f1**	installation dialog
2	**ctrl**+**alt**+**f2**	shell prompt
3	**ctrl**+**alt**+**f3**	install log (messages from installation program)
4	**ctrl**+**alt**+**f4**	system-related messages
5	**ctrl**+**alt**+**f5**	other messages
6	**ctrl**+**alt**+**f6**	x graphical display

4.2. Screenshots during installation

Anaconda allows you to take screenshots during the installation process. At any time during installation, press **Shift**+**Print Screen** and **anaconda** will save a screenshot to **/root/anaconda-screenshots**.

If you are performing a Kickstart installation, use the **autostep --autoscreenshot** option to generate a screenshot of each step of the installation automatically. Refer to Section 31.3, “Creating the Kickstart File” for details of configuring a Kickstart file.

4.3. The Text Mode Installation Program User Interface

The Red Hat Enterprise Linux text mode installation program uses a screen-based interface that

includes most of the on-screen *widgets* commonly found on graphical user interfaces. Figure 4.1, "Installation Program Widgets as seen in **Boot Loader Configuration**", and Figure 4.2, "Installation Program Widgets as seen in Disk Druid", illustrate the screens that appear during the installation process.

Note

While text mode installations are not explicitly documented, those using the text mode installation program can easily follow the GUI installation instructions. One thing to note is that manipulation of LVM (Logical Volume Management) disk volumes is only possible in graphical mode. In text mode it is only possible to view and accept the default LVM setup.

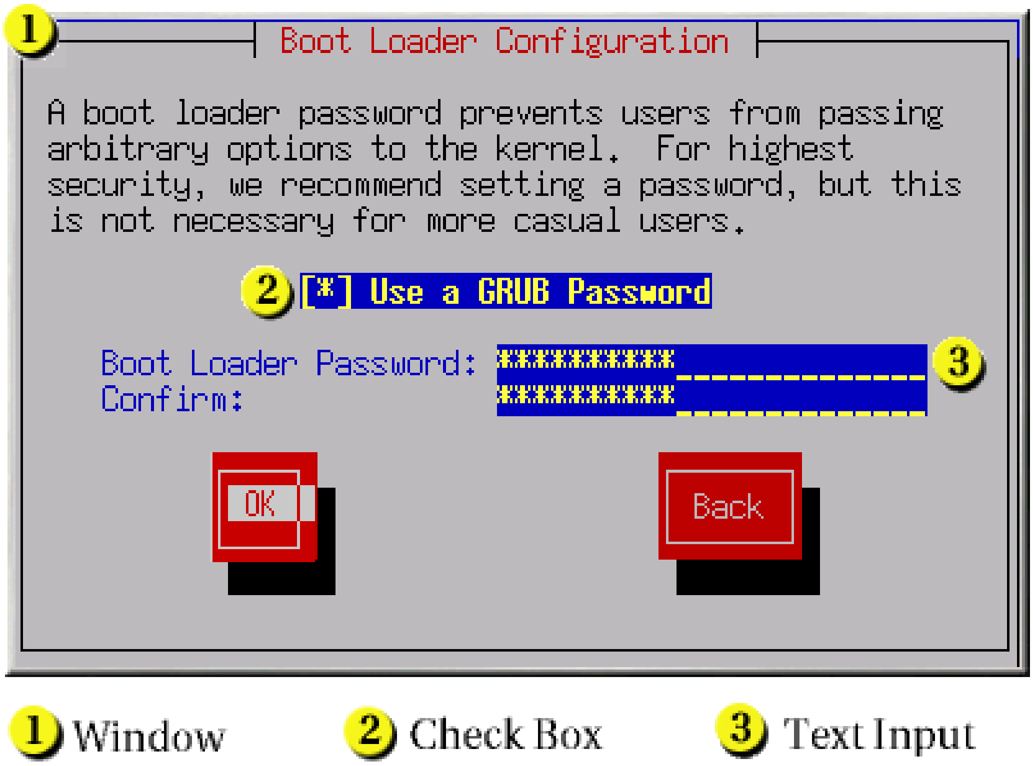

Figure 4.1. Installation Program Widgets as seen in Boot Loader Configuration

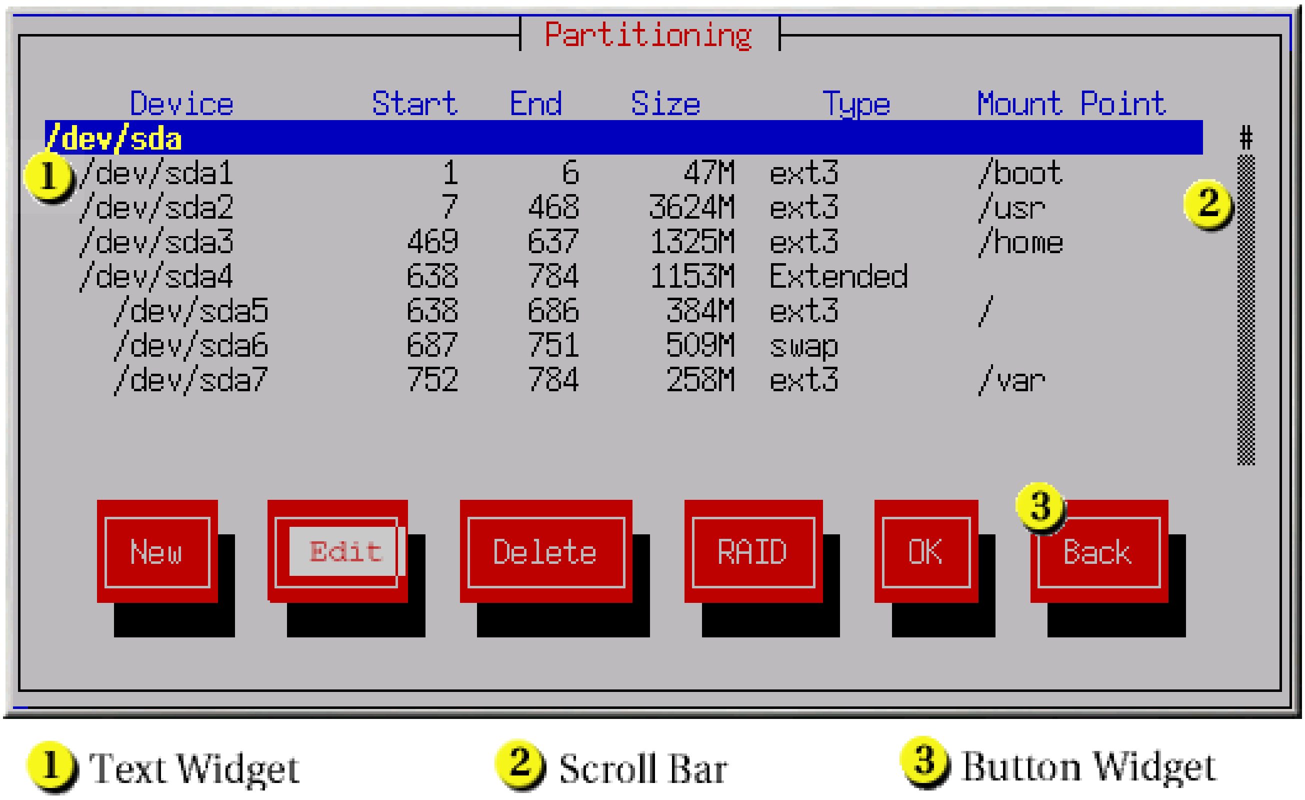

Figure 4.2. Installation Program Widgets as seen in Disk Druid

Here is a list of the most important widgets shown in Figure 4.1, "Installation Program Widgets as seen in **Boot Loader Configuration**" and Figure 4.2, "Installation Program Widgets as seen in Disk Druid":

- Window — Windows (usually referred to as *dialogs* in this manual) appear on your screen throughout the installation process. At times, one window may overlay another; in these cases, you can only interact with the window on top. When you are finished in that window, it disappears, allowing you to continue working in the window underneath.

- Checkbox — Checkboxes allow you to select or deselect a feature. The box displays either an asterisk (selected) or a space (unselected). When the cursor is within a checkbox, press **Space** to select or deselect a feature.

- Text Input — Text input lines are regions where you can enter information required by the installation program. When the cursor rests on a text input line, you may enter and/or edit information on that line.

- Text Widget — Text widgets are regions of the screen for the display of text. At times, text widgets may also contain other widgets, such as checkboxes. If a text widget contains more information than can be displayed in the space reserved for it, a scroll bar appears; if you position the cursor within the text widget, you can then use the **Up** and **Down** arrow keys to scroll through all the information available. Your current position is shown on the scroll bar by a # character, which moves up and down the scroll bar as you scroll.

- Scroll Bar — Scroll bars appear on the side or bottom of a window to control which part of a list or document is currently in the window's frame. The scroll bar makes it easy to move to any part of a file.

- Button Widget — Button widgets are the primary method of interacting with the installation program. You progress through the windows of the installation program by navigating these buttons, using the **Tab** and **Enter** keys. Buttons can be selected when they are highlighted.

- Cursor — Although not a widget, the cursor is used to select (and interact with) a particular widget. As the cursor is moved from widget to widget, it may cause the widget to change color, or the cursor itself may only appear positioned in or next to the widget.

4.3.1. Using the Keyboard to Navigate

Navigation through the installation dialogs is performed through a simple set of keystrokes. To move the cursor, use the **Left**, **Right**, **Up**, and **Down** arrow keys. Use **Tab**, and **Shift**-**Tab** to cycle forward or backward through each widget on the screen. Along the bottom, most screens display a summary of available cursor positioning keys.

To "press" a button, position the cursor over the button (using **Tab**, for example) and press **Space** or **Enter**. To select an item from a list of items, move the cursor to the item you wish to select and press **Enter**. To select an item with a checkbox, move the cursor to the checkbox and press **Space** to select an item. To deselect, press **Space** a second time.

Pressing **F12** accepts the current values and proceeds to the next dialog; it is equivalent to pressing the **OK** button.

> **Warning**
>
> Unless a dialog box is waiting for your input, do not press any keys during the installation process (doing so may result in unpredictable behavior).

4.4. Starting the Installation Program

To start, first make sure that you have all necessary resources for the installation. If you have already read through Chapter 2, *Steps to Get You Started*, and followed the instructions, you should be ready to start the installation process. When you have verified that you are ready to begin, boot the installation program using the Red Hat Enterprise Linux DVD or CD-ROM #1 or any boot media that you have created.

> **Note**
>
> Occasionally, some hardware components require a *driver diskette* during the installation. A driver diskette adds support for hardware that is not otherwise supported by the installation program. Refer to Chapter 7, *Updating drivers during installation on Intel and AMD systems* for more information.

4.4.1. Booting the Installation Program on x86, AMD64, and Intel® 64 Systems

You can boot the installation program using any one of the following media (depending upon what your system can support):

- *Red Hat Enterprise Linux DVD/CD-ROM* — Your machine supports a bootable DVD/CD-ROM drive and you have the Red Hat Enterprise Linux CD-ROM set or DVD.

- *Boot CD-ROM* — Your machine supports a bootable CD-ROM drive and you want to perform network or hard drive installation.

- *USB pen drive* — Your machine supports booting from a USB device.
- *PXE boot via network* — Your machine supports booting from the network. This is an advanced installation path. Refer to Chapter 34, *PXE Network Installations* for additional information on this method.

To create a boot CD-ROM or to prepare your USB pen drive for installation, refer to Section 2.4.1, “Alternative Boot Methods”.

Insert the boot media and reboot the system. Your BIOS settings may need to be changed to allow you to boot from the CD-ROM or USB device.

Note

> To change your BIOS settings on an x86, AMD64, or Intel® 64 system, watch the instructions provided on your display when your computer first boots. A line of text appears, telling you which key to press to enter the BIOS settings.
>
> Once you have entered your BIOS setup program, find the section where you can alter your boot sequence. The default is often C, A or A, C (depending on whether you boot from your hard drive [C] or a diskette drive [A]). Change this sequence so that the CD-ROM is first in your boot order and that C or A (whichever is your typical boot default) is second. This instructs the computer to first look at the CD-ROM drive for bootable media; if it does not find bootable media on the CD-ROM drive, it then checks your hard drive or diskette drive.
>
> Save your changes before exiting the BIOS. For more information, refer to the documentation that came with your system.

After a short delay, a screen containing the **`boot:`** prompt should appear. The screen contains information on a variety of boot options. Each boot option also has one or more help screens associated with it. To access a help screen, press the appropriate function key as listed in the line at the bottom of the screen.

As you boot the installation program, be aware of two issues:

- Once the **`boot:`** prompt appears, the installation program automatically begins if you take no action within the first minute. To disable this feature, press one of the help screen function keys.
- If you press a help screen function key, there is a slight delay while the help screen is read from the boot media.

Normally, you only need to press **`Enter`** to boot. Be sure to watch the boot messages to review if the Linux kernel detects your hardware. If your hardware is properly detected, continue to the next section. If it does not properly detect your hardware, you may need to restart the installation and use one of the boot options provided in Chapter 8, *Additional Boot Options for Intel® and AMD Systems*.

4.4.2. Booting the Installation Program on Itanium Systems

Your Itanium system should be able to boot the Red Hat Enterprise Linux installation program directly from the Red Hat Enterprise Linux CD #1. If your Itanium cannot boot the installation program from the CD-ROM (or if you want to perform a hard drive, NFS, FTP, or HTTP installation) you must boot from an LS-120 diskette. Refer to Section 4.4.2.2, “Booting the Installation Program from an LS-120 Diskette” for more information.

4.4.2.1. Booting the Installation Program from the DVD/CD-ROM

To boot from the Red Hat Enterprise Linux CD #1 follow these steps:

1. Remove all media except Red Hat Enterprise Linux CD #1.
2. From the **Boot Option** menu choose **EFI Shell**.
3. At the **`Shell>`** prompt, change to the file system on the CD-ROM. For example, in the above sample **`map`** output, the system partition on the CD-ROM is **`fs1`**. To change to the **`fs1`** file system, type **`fs1:`** at the prompt.
4. Type **`elilo linux`** to boot into the installation program.
5. Go to Chapter 4, *Installing on Intel® and AMD Systems* to begin the installation.

4.4.2.2. Booting the Installation Program from an LS-120 Diskette

If your Itanium cannot boot from Red Hat Enterprise Linux CD #1, you must boot from an LS-120 diskette. If you want to perform a hard drive, NFS, FTP, or HTTP installation, you must boot from a boot LS-120 diskette.

You must create an LS-120 boot image file diskette from the boot image file on CD #1: **`images/boot.img`**. To create this diskette in Linux, insert a blank LS-120 diskette and type the following command at a shell prompt:

```
dd if=boot.img of=/dev/hda bs=180k
```

Replace *boot.img* with the full path to the boot image file and **`/dev/hda`** with the correct device name for the LS-120 diskette drive.

If you are not using the Red Hat Enterprise Linux CD, the installation program starts in text mode and you must choose a few basic options for your system.

If you are using the CD-ROM to load the installation program, follow the instructions contained in Chapter 4, *Installing on Intel® and AMD Systems*.

To boot from an LS-120 diskette follow these steps:

1. Insert the LS-120 diskette you made from the boot image file **`boot.img`**. If you are performing a local CD-ROM installation but booting off the LS-120 diskette, insert the Red Hat Enterprise Linux CD #1 also. If you are performing a hard drive, NFS, FTP, or HTTP installation, you do not need the CD-ROM.
2. From the **Boot Option** menu choose **EFI Shell**.
3. At the **`Shell>`** prompt, change the device to the LS-120 drive by typing the command **`fs0:`**, using the example **`map`** output above.
4. Type **`elilo linux`** to boot into the installation program.
5. Go to Chapter 4, *Installing on Intel® and AMD Systems* to begin the installation.

4.4.3. Additional Boot Options

While it is easiest to boot using a CD-ROM and perform a graphical installation, sometimes there are installation scenarios where booting in a different manner may be needed. This section discusses additional boot options available for Red Hat Enterprise Linux.

For Itanium users:

To pass options to the boot loader on an Itanium system, enter the following at the EFI Shell prompt:

```
elilo linux option
```

For x86, AMD64, and Intel® 64 users:

To pass options to the boot loader on an x86, AMD64, or Intel® 64 system, use the instructions as provided in the boot loader option samples below.

> **Note**
>
> Refer to Chapter 8, *Additional Boot Options for Intel® and AMD Systems* for additional boot options not covered in this section.

- To perform a text mode installation, at the installation boot prompt, type:

  ```
  linux text
  ```

- ISO images have an md5sum embedded in them. To test the checksum integrity of an ISO image, at the installation boot prompt, type:

  ```
  linux mediacheck
  ```

  The installation program prompts you to insert a CD or select an ISO image to test, and select **OK** to perform the checksum operation. This checksum operation can be performed on any Red Hat Enterprise Linux CD and does not have to be performed in a specific order (for example, CD #1 does not have to be the first CD you verify). It is strongly recommended to perform this operation on any Red Hat Enterprise Linux CD that was created from downloaded ISO images. This command works with the CD, DVD, hard drive ISO, and NFS ISO installation methods.

- Also in the **images/** directory is the **boot.iso** file. This file is an ISO image than can be used to boot the installation program. To use the **boot.iso**, your computer must be able to boot from its CD-ROM drive, and its BIOS settings must be configured to do so. You must then burn the **boot.iso** file onto a recordable/rewriteable CD-ROM.

- If you need to perform the installation in *serial mode*, type the following command:

  ```
  linux console=<device>
  ```

  For text mode installations, use:

  ```
  linux text console=<device>
  ```

  In the above command, *<device>* should be the device you are using (such as ttyS0 or ttyS1). For example, **linux text console=ttyS0**.

  Text mode installations using a serial terminal work best when the terminal supports UTF-8. Under UNIX and Linux, Kermit supports UTF-8. For Windows, Kermit '95 works well. Non-UTF-8 capable terminals works as long as only English is used during the installation process. An enhanced serial display can be used by passing the **utf8** command as a boot-time option to the installation program. For example:

```
linux console=ttyS0 utf8
```

4.4.3.1. Kernel Options

Options can also be passed to the kernel. For example, to apply updates for the anaconda installation program from a floppy disk enter:

```
linux updates
```

For text mode installations, use:

```
linux text updates
```

This command will prompt you to insert a floppy diskette containing updates for anaconda. It is not needed if you are performing a network installation and have already placed the updates image contents in **rhupdates/** on the server.

After entering any options, press **Enter** to boot using those options.

If you need to specify boot options to identify your hardware, please write them down. The boot options are needed during the boot loader configuration portion of the installation (refer to Section 4.20, “x86, AMD64, and Intel® 64 Boot Loader Configuration” for more information).

For more information on kernel options refer to Chapter 8, *Additional Boot Options for Intel® and AMD Systems*.

4.5. Selecting an Installation Method

What type of installation method do you wish to use? The following installation methods are available:

DVD/CD-ROM

If you have a DVD/CD-ROM drive and the Red Hat Enterprise Linux CD-ROMs or DVD you can use this method. Refer to Section 4.6, “Installing from DVD/CD-ROM”, for DVD/CD-ROM installation instructions.

Hard Drive

If you have copied the Red Hat Enterprise Linux ISO images to a local hard drive, you can use this method. You need a boot CD-ROM (use the **linux askmethod** boot option). Refer to Section 4.7, “Installing from a Hard Drive”, for hard drive installation instructions.

NFS

If you are installing from an NFS server using ISO images or a mirror image of Red Hat Enterprise Linux, you can use this method. You need a boot CD-ROM (use the **linux askmethod** boot option). Refer to Section 4.9, “Installing via NFS” for network installation instructions. Note that NFS installations may also be performed in GUI mode.

FTP

If you are installing directly from an FTP server, use this method. You need a boot CD-ROM (use the **linux askmethod** boot option). Refer to Section 4.10, “Installing via FTP”, for FTP installation instructions.

HTTP

If you are installing directly from an HTTP (Web) server, use this method. You need a boot CD-ROM (use the **`linux askmethod`** boot option). Refer to Section 4.11, “Installing via HTTP”, for HTTP installation instructions.

4.6. Installing from DVD/CD-ROM

To install Red Hat Enterprise Linux from a DVD/CD-ROM, place the DVD or CD #1 in your DVD/CD-ROM drive and boot your system from the DVD/CD-ROM.

The installation program then probes your system and attempts to identify your CD-ROM drive. It starts by looking for an IDE (also known as an ATAPI) CD-ROM drive.

Note

To abort the installation process at this time, reboot your machine and then eject the boot media. You can safely cancel the installation at any point before the **`About to Install`** screen. Refer to Section 4.25, “Preparing to Install” for more information.

If your CD-ROM drive is not detected, and it is a SCSI CD-ROM, the installation program prompts you to choose a SCSI driver. Choose the driver that most closely resembles your adapter. You may specify options for the driver if necessary; however, most drivers detect your SCSI adapter automatically.

If the DVD/CD-ROM drive is found and the driver loaded, the installer will present you with the option to perform a media check on the DVD/CD-ROM. This will take some time, and you may opt to skip over this step. However, if you later encounter problems with the installer, you should reboot and perform the media check before calling for support. From the media check dialog, continue to the next stage of the installation process (refer to Section 4.12, “Welcome to Red Hat Enterprise Linux”).

4.7. Installing from a Hard Drive

The **`Select Partition`** screen applies only if you are installing from a disk partition (that is, if you used the **`askmethod`** boot options and selected **Hard Drive** in the **`Installation Method`** dialog). This dialog allows you to name the disk partition and directory from which you are installing Red Hat Enterprise Linux. If you used the **`repo=hd`** boot option, you already specified a partition.

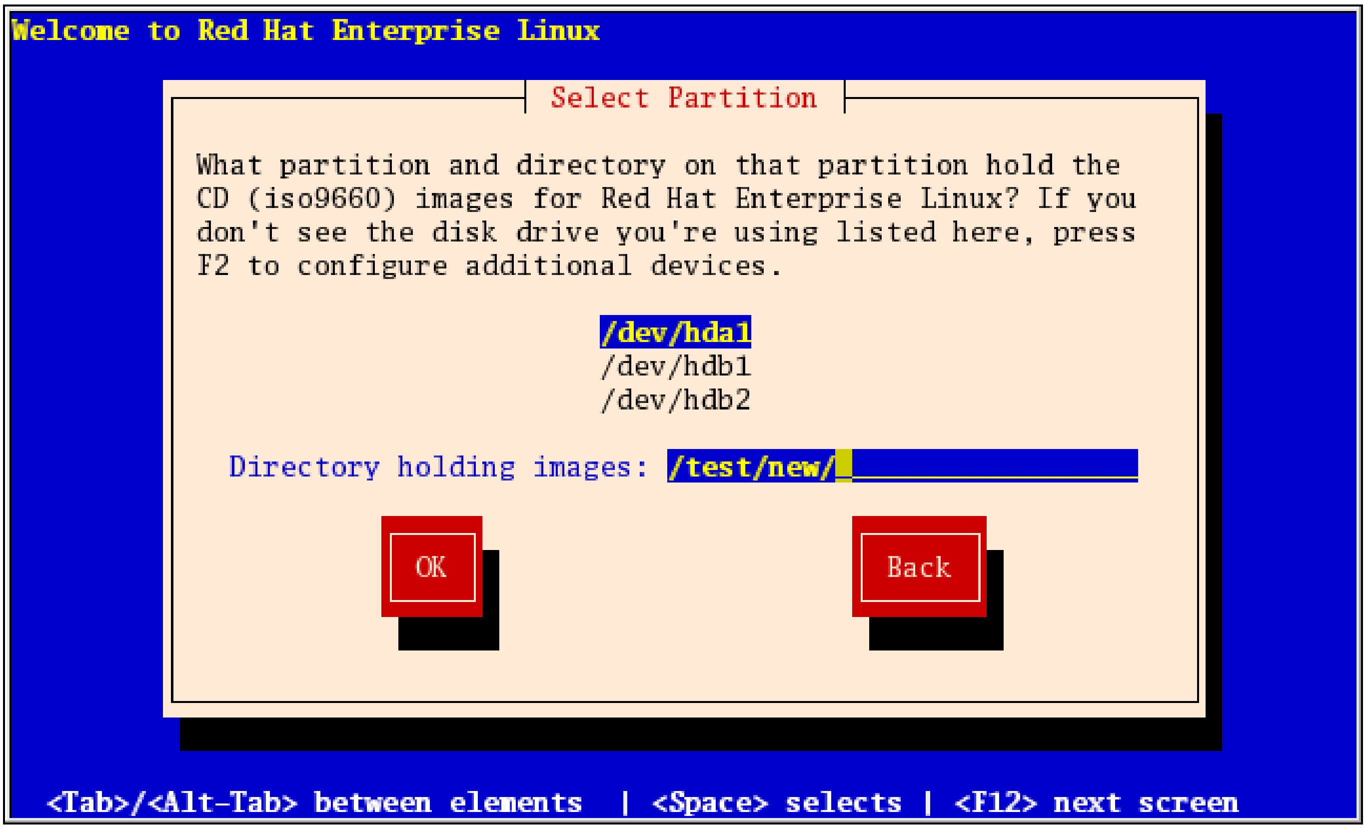

Figure 4.3. Selecting Partition Dialog for Hard Drive Installation

Enter the device name of the partition containing the Red Hat Enterprise Linux ISO images. This partition must be formatted with a ext2 or vfat filesystem, and cannot be a logical volume. There is also a field labeled **`Directory holding images`**.

If the ISO images are in the root (top-level) directory of a partition, enter a `/`. If the ISO images are located in a subdirectory of a mounted partition, enter the name of the directory holding the ISO images within that partition. For example, if the partition on which the ISO images is normally mounted as **`/home/`**, and the images are in **`/home/new/`**, you would enter **`/new/`**.

After you have identified the disk partition, the **`Welcome`** dialog appears.

4.8. Performing a Network Installation

If you are performing a network installation and booted with the **`askmethod`** boot option, the **`Configure TCP/IP`** dialog appears. This dialog asks for your IP and other network addresses. You can choose to configure the IP address and Netmask of the device via DHCP or manually. If manually, you have the option to enter IPv4 and/or IPv6 information. Enter the IP address you are using during installation and press **`Enter`**. Note that you need to supply IPv4 information if you wish to perform an NFS installation.

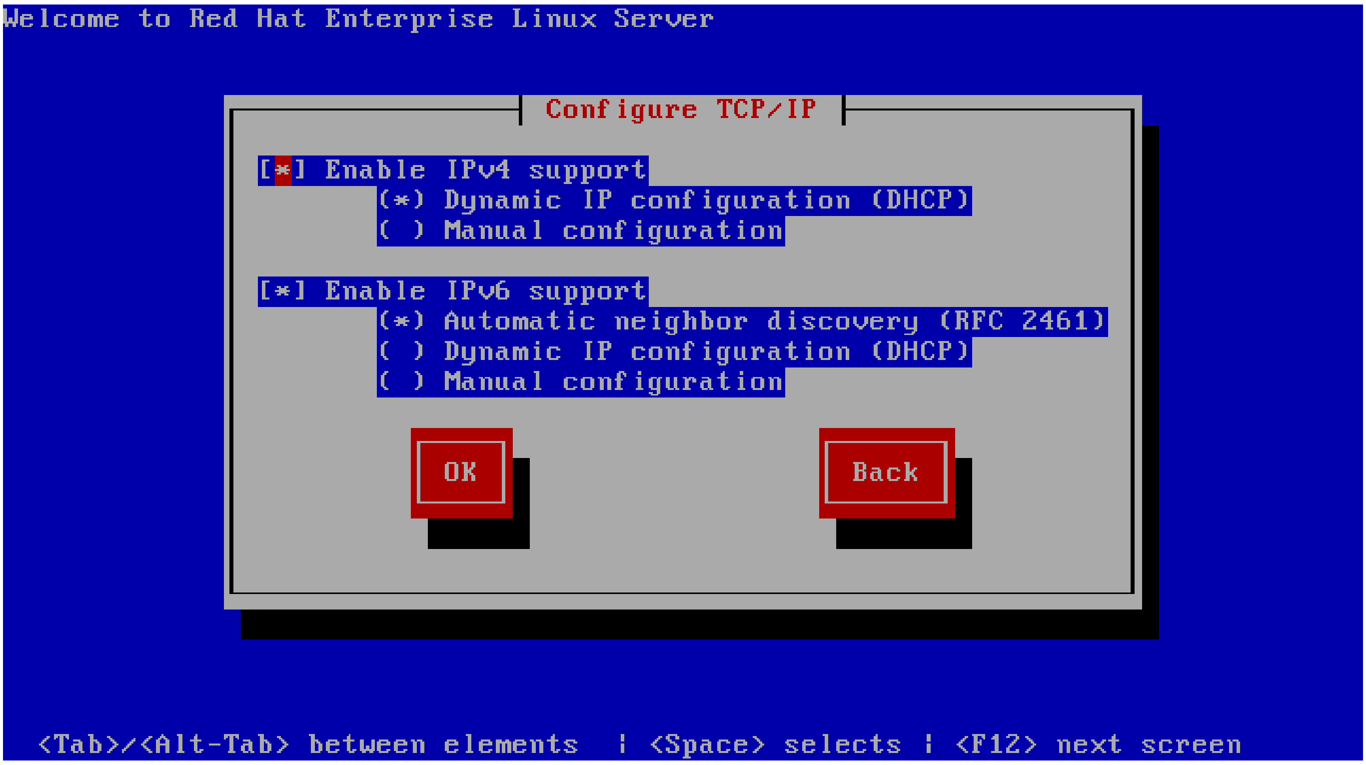

Figure 4.4. TCP/IP Configuration

4.9. Installing via NFS

The NFS dialog applies only if you are installing from an NFS server (if you selected **NFS Image** in the **`Installation Method`** dialog).

Enter the domain name or IP address of your NFS server. For example, if you are installing from a host named **`eastcoast`** in the domain **`example.com`**, enter **`eastcoast.example.com`** in the **`NFS Server`** field.

Next, enter the name of the exported directory. If you followed the setup described in Section 2.5, “Preparing for a Network Installation”, you would enter the directory ***`/export/directory/`***.

If the NFS server is exporting a mirror of the Red Hat Enterprise Linux installation tree, enter the directory which contains the root of the installation tree. You will enter an Installation Key later on in the process which will determine which subdirectories are used to install from. If everything was specified properly, a message appears indicating that the installation program for Red Hat Enterprise Linux is running.

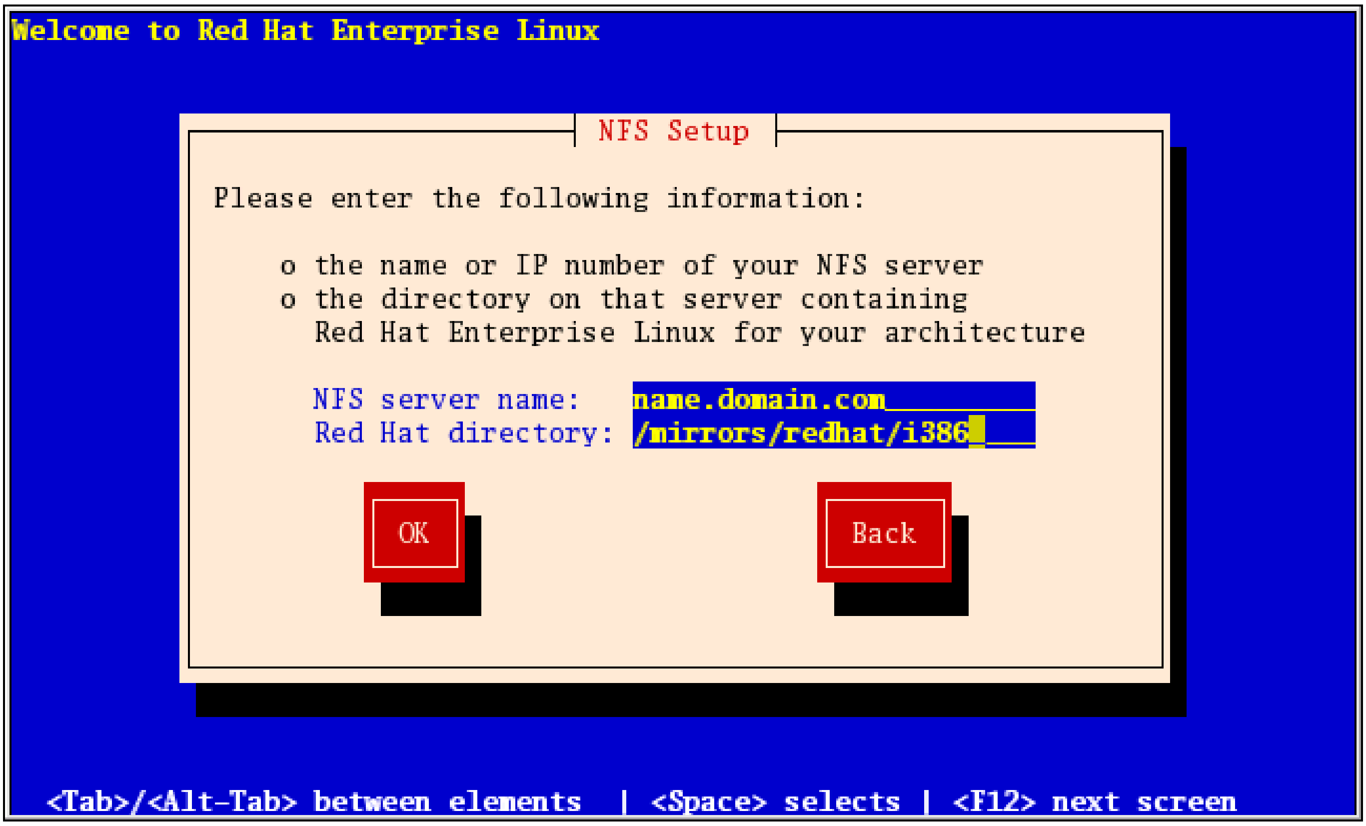

Figure 4.5. NFS Setup Dialog

If the NFS server is exporting the ISO images of the Red Hat Enterprise Linux CD-ROMs, enter the directory which contains the ISO images.

Next, the **`Welcome`** dialog appears.

4.10. Installing via FTP

The FTP dialog applies only if you are installing from an FTP server (that is, if you used the **`askmethod`** boot options and selected **FTP** in the **`Installation Method`** dialog). This dialog allows you to identify the FTP server from which you are installing Red Hat Enterprise Linux. If you used the **`repo=ftp`** boot option, you already specified a server and path.

Figure 4.6. FTP Setup Dialog

Enter the name or IP address of the FTP site you are installing from, and the name of the directory containing the ***`variant/`*** directory for your architecture. For example, if the FTP site contains the directory **`/mirrors/redhat/`*`arch`*`/`*`variant`*`;/`**, enter **`/mirrors/redhat/`*`arch`*`/`** (where *arch* is replaced with the architecture type of your system, such as i386, ia64, ppc, or s390x, and *variant* is the variant that you are installing, such as Client, Server, Workstation, etc.). If everything was specified properly, a message box appears indicating that files are being retrieved from the server.

Next, the **`Welcome`** dialog appears.

Note

You can save disk space by using the ISO images you have already copied to the server. To accomplish this, install Red Hat Enterprise Linux using ISO images without copying them into a single tree by loopback mounting them. For each ISO image:

`mkdir disc`*`X`*

`mount -o loop RHEL5-disc`*`X`*`.iso disc`*`X`*

Replace **X** with the corresponding disc number.

4.11. Installing via HTTP

The HTTP dialog applies only if you are installing from an HTTP server (that is, if you used the **`askmethod`** boot option and selected **`HTTP`** in the **`Installation Method`** dialog). This dialog prompts you for information about the HTTP server from which you are installing Red Hat Enterprise Linux. If you used the **`repo=http`** boot option, you already specified a server and path.

Enter the name or IP address of the HTTP site you are installing from, and the name of the directory containing the ***`variant/`*** directory for your architecture. For example, if the HTTP site contains the directory **`/mirrors/redhat/`*`arch`*`/`*`variant`*`/`**, enter **`/mirrors/redhat/`*`arch`*`/`** (where *arch* is replaced with the architecture type of your system, such as i386, ia64, ppc, or s390x, and *variant* is the variant that you are installing, such as Client, Server, Workstation, etc.). If everything was specified properly, a message box appears indicating that files are being retrieved from the server.

Figure 4.7. HTTP Setup Dialog

Next, the **`Welcome`** dialog appears.

Note

You can save disk space by using the ISO images you have already copied to the server. To accomplish this, install Red Hat Enterprise Linux using ISO images without copying them into a single tree by loopback mounting them. For each ISO image:

`mkdir disc`*`X`*

`mount -o loop RHEL5-disc`*`X`*`.iso disc`*`X`*

Replace **X** with the corresponding disc number.

4.12. Welcome to Red Hat Enterprise Linux

The **`Welcome`** screen does not prompt you for any input. From this screen you can access the Release Notes for Red Hat Enterprise Linux 5.11 by clicking on the Release Notes button.

Click on the **`Next`** button to continue.

4.13. Language Selection

Using your mouse, select a language to use for the installation (refer to Figure 4.8, “Language Selection”).

The language you select here will become the default language for the operating system once it is installed. Selecting the appropriate language also helps target your time zone configuration later in the installation. The installation program tries to define the appropriate time zone based on what you specify on this screen.

Figure 4.8. Language Selection

Once you select the appropriate language, click **Next** to continue.

4.14. Keyboard Configuration

Using your mouse, select the correct layout type (for example, U.S. English) for the keyboard you would prefer to use for the installation and as the system default (refer to the figure below).

Once you have made your selection, click **Next** to continue.

Figure 4.9. Keyboard Configuration

> **Note**
>
> To change your keyboard layout type after you have completed the installation, use the **Keyboard Configuration Tool**.
>
> Type the **`system-config-keyboard`** command in a shell prompt to launch the **Keyboard Configuration Tool**. If you are not root, it prompts you for the root password to continue.

4.15. Enter the Installation Number

Enter your Installation Number (refer to Figure 4.10, “Installation Number”). This number will determine the package selection set that is available to the installer. If you choose to skip entering the installation number you will be presented with a basic selection of packages to install later on.

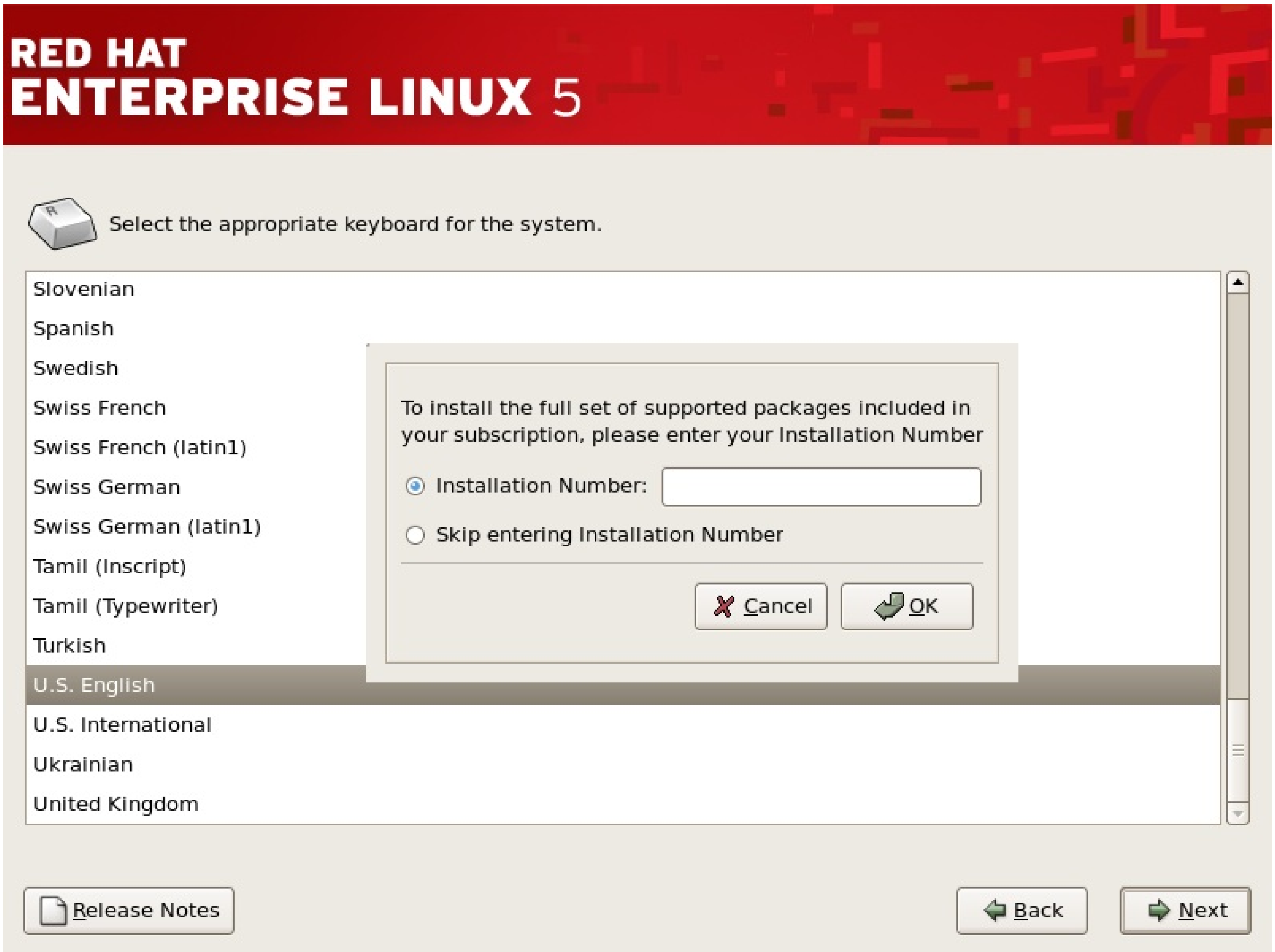

Figure 4.10. Installation Number

4.16. Disk Partitioning Setup

Partitioning allows you to divide your hard drive into isolated sections, where each section behaves as its own hard drive. Partitioning is particularly useful if you run multiple operating systems. If you are not sure how you want your system to be partitioned, read Chapter 26, *An Introduction to Disk Partitions* for more information.

On this screen you can choose to create the default layout or choose to manual partition using the 'Create custom layout' option of **Disk Druid**.

The first three options allow you to perform an automated installation without having to partition your drive(s) yourself. If you do not feel comfortable with partitioning your system, it is recommended that you *do not* choose to create a custom layout and instead let the installation program partition for you.

You can configure an iSCSI target for installation, or disable a dmraid device from this screen by clicking on the 'Advanced storage configuration' button. For more information refer to Section 4.17, " Advanced Storage Options ".

> **Warning**
>
> The **Update Agent** downloads updated packages to **`/var/cache/yum/`** by default. If you partition the system manually, and create a separate **`/var/`** partition, be sure to create the partition large enough (3.0 GB or more) to download package updates.

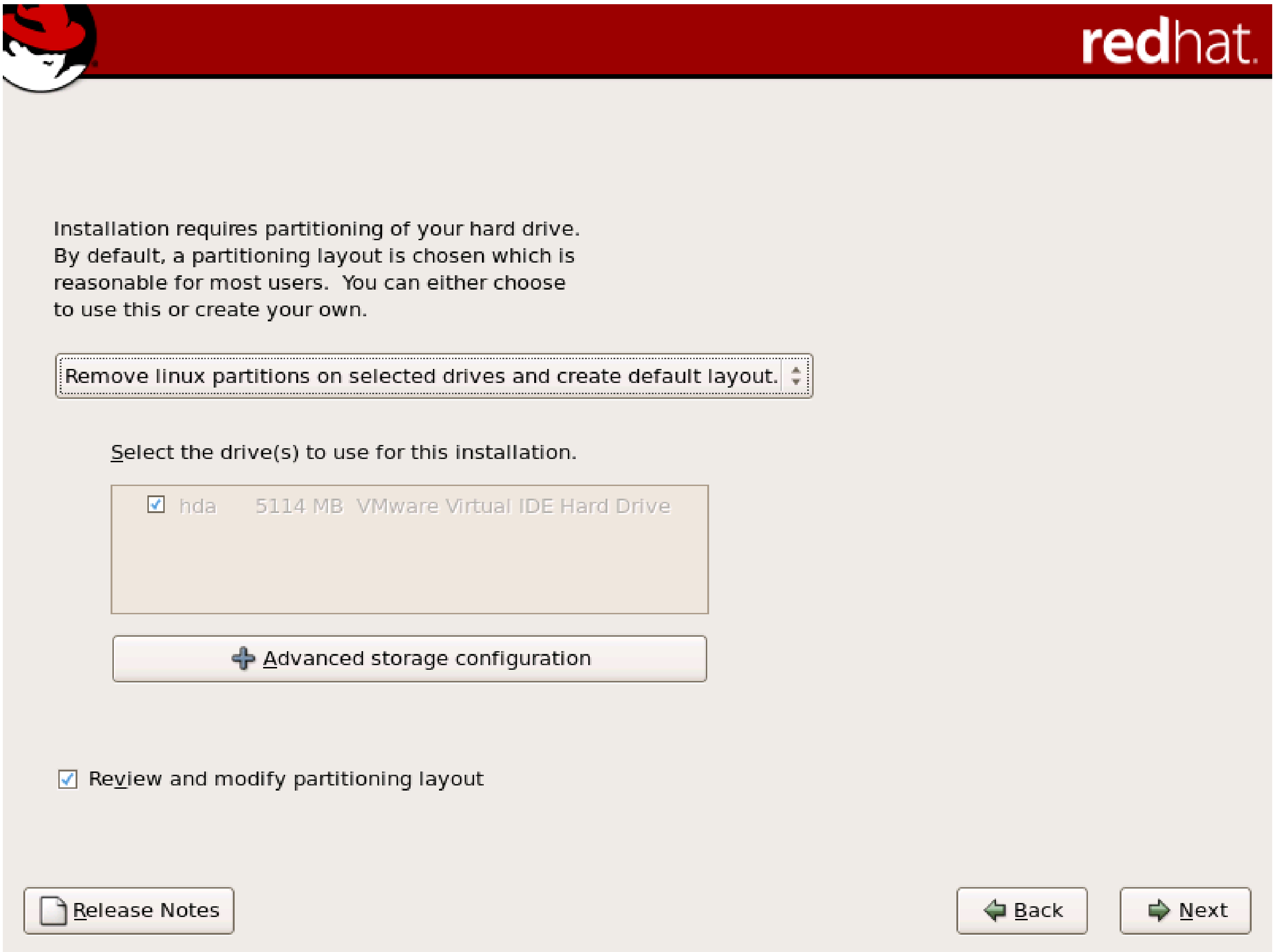

Figure 4.11. Disk Partitioning Setup

If you choose to create a custom layout using **Disk Druid**, refer to Section 4.19, “Partitioning Your System”.

Warning

If you receive an error after the **`Disk Partitioning Setup`** phase of the installation saying something similar to:

"The partition table on device hda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive."

you may not have a partition table on that drive or the partition table on the drive may not be recognizable by the partitioning software used in the installation program.

Users who have used programs such as **EZ-BIOS** have experienced similar problems, causing data to be lost (assuming the data was not backed up before the installation began).

No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

Important — multipath devices

To install Red Hat Enterprise Linux on a network device accessible through multiple paths, deselect all local storage in the **`Select the drive(s) to use for this installation`** window, and select a device labeled **`mapper/mpath`** instead.

Note that migrating the root file system of an existing Red Hat Enterprise Linux installation from single path storage to multipath storage is not supported. You must perform a new installation to move the root file system to a multipath storage device. Therefore you should plan your installation accordingly. See https://access.redhat.com/site/solutions/66501 for more information.

4.17. Advanced Storage Options

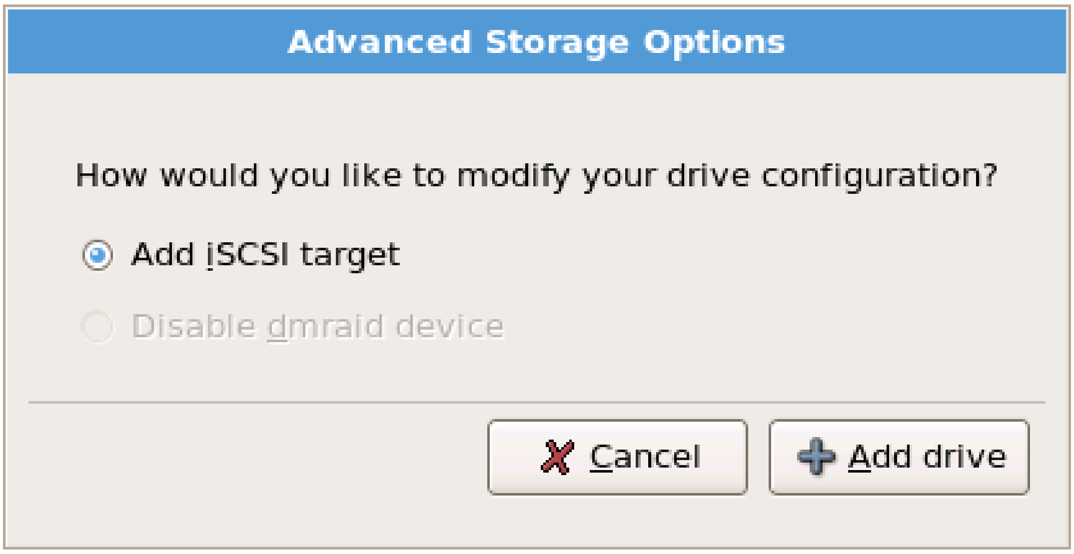

Figure 4.12. Advanced Storage Options

From this screen you can choose to disable a dmraid device, in which case the individual elements of the dmraid device will appear as separate hard drives. You can also choose to configure an iSCSI (SCSI over TCP/IP) target.

To configure an ISCSI target invoke the **`Configure ISCSI Parameters`** dialog by selecting **`Add ISCSI target`** and clicking on the **`Add Drive`** button. If a network connection is not already active, the installer prompts you to provide details of your network interface. Select your network interface from the drop-down menu, then either leave the **`Use dynamic IP configuration`** box checked, or uncheck it to enter the IP address of your system and the IP addresses of the gateway and nameserver on your network. Ensure that the **`Enable IPv4`** box remains checked.

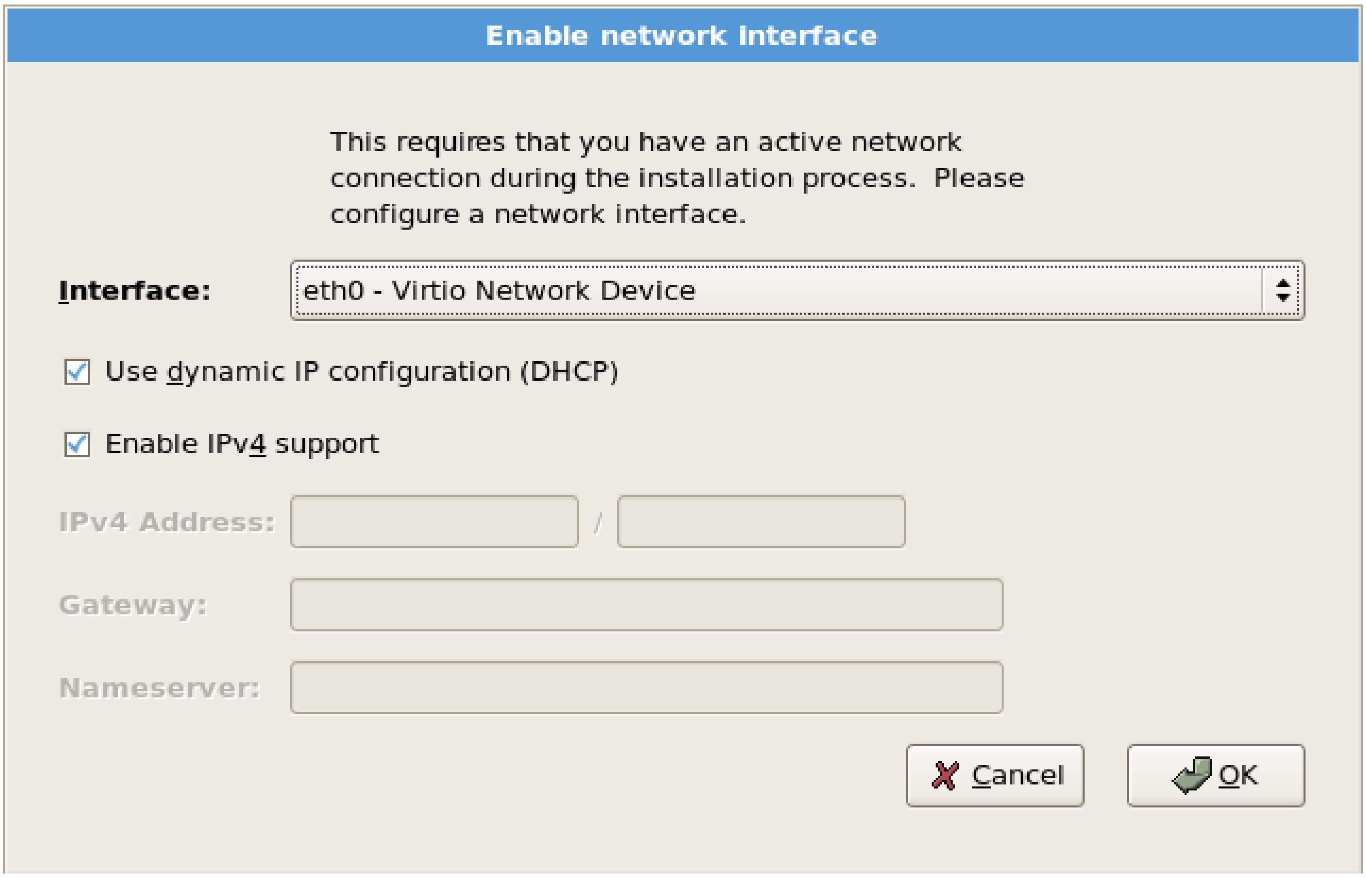

Figure 4.13. Enable network Interface

Fill in the details for the iSCSI target IP and provide a unique iSCSI initiator name to identify this system. If the iSCSI target uses *Challenge Handshake Authentication Protocol* (CHAP) for authentication, enter the CHAP username and password. If your environment uses 2-way CHAP (also called "Mutual CHAP"), also enter the reverse CHAP username and password. Click the **`Add target`** button to attempt connection to the ISCSI target using this information.

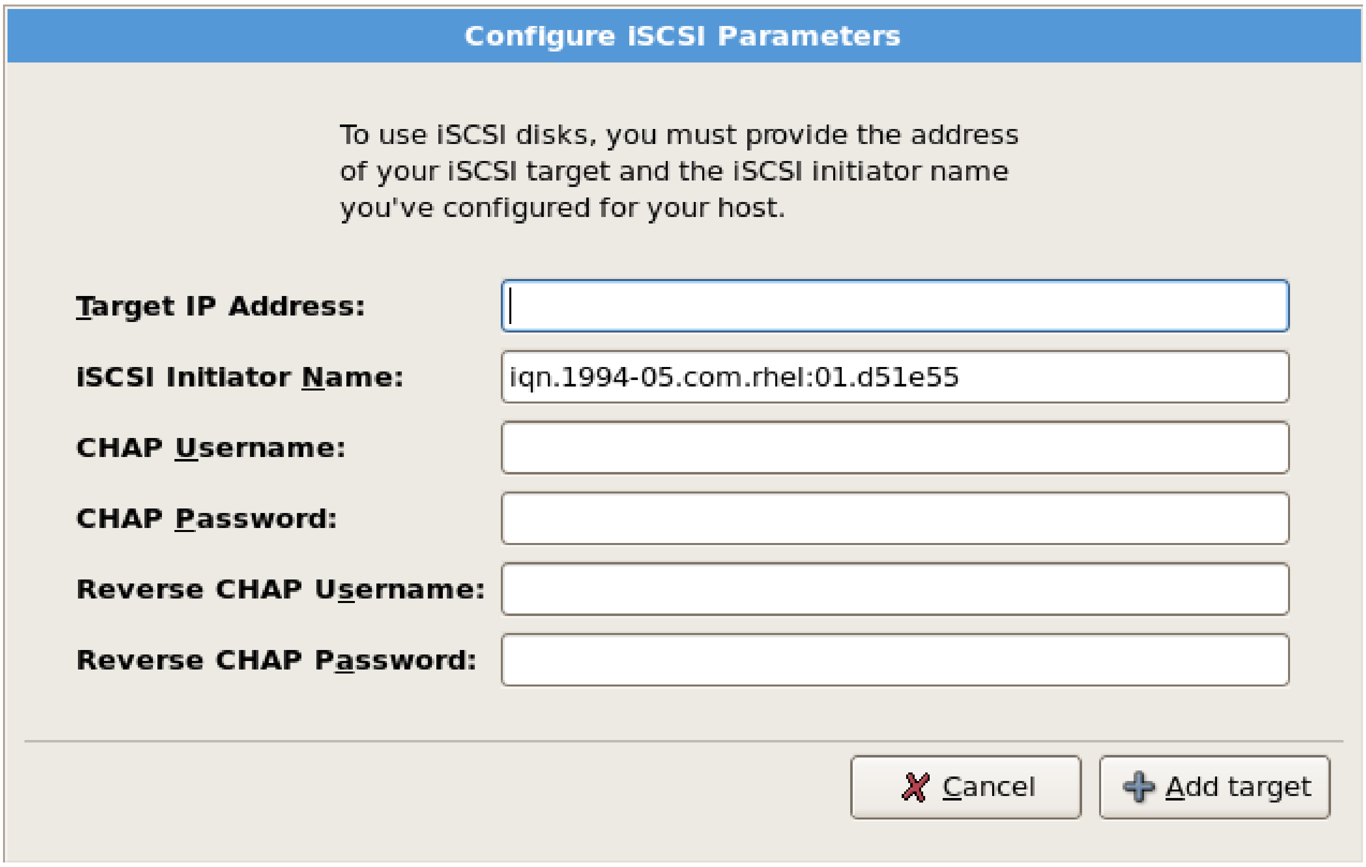

Figure 4.14. Configure ISCSI Parameters

Please note that you will be able to reattempt with a different ISCSI target IP should you enter it incorrectly, but in order to change the ISCSI initiator name you will need to restart the installation.

4.18. Create Default Layout

Create default layout allows you to have some control concerning what data is removed (if any) from your system. Your options are:

- **`Remove all partitions on selected drives and create default layout`** — select this option to remove all partitions on your hard drive(s) (this includes partitions created by other operating systems such as Windows VFAT or NTFS partitions).

> **Warning**
>
> If you select this option, all data on the selected hard drive(s) is removed by the installation program. Do not select this option if you have information that you want to keep on the hard drive(s) where you are installing Red Hat Enterprise Linux.

- **`Remove Linux partitions on selected drives and create default layout`** — select this option to remove only Linux partitions (partitions created from a previous Linux installation). This does not remove other partitions you may have on your hard drive(s) (such as VFAT or FAT32 partitions).

- **`Use free space on selected drives and create default layout`** — select this option to retain your current data and partitions, assuming you have enough free space available on your hard drive(s).

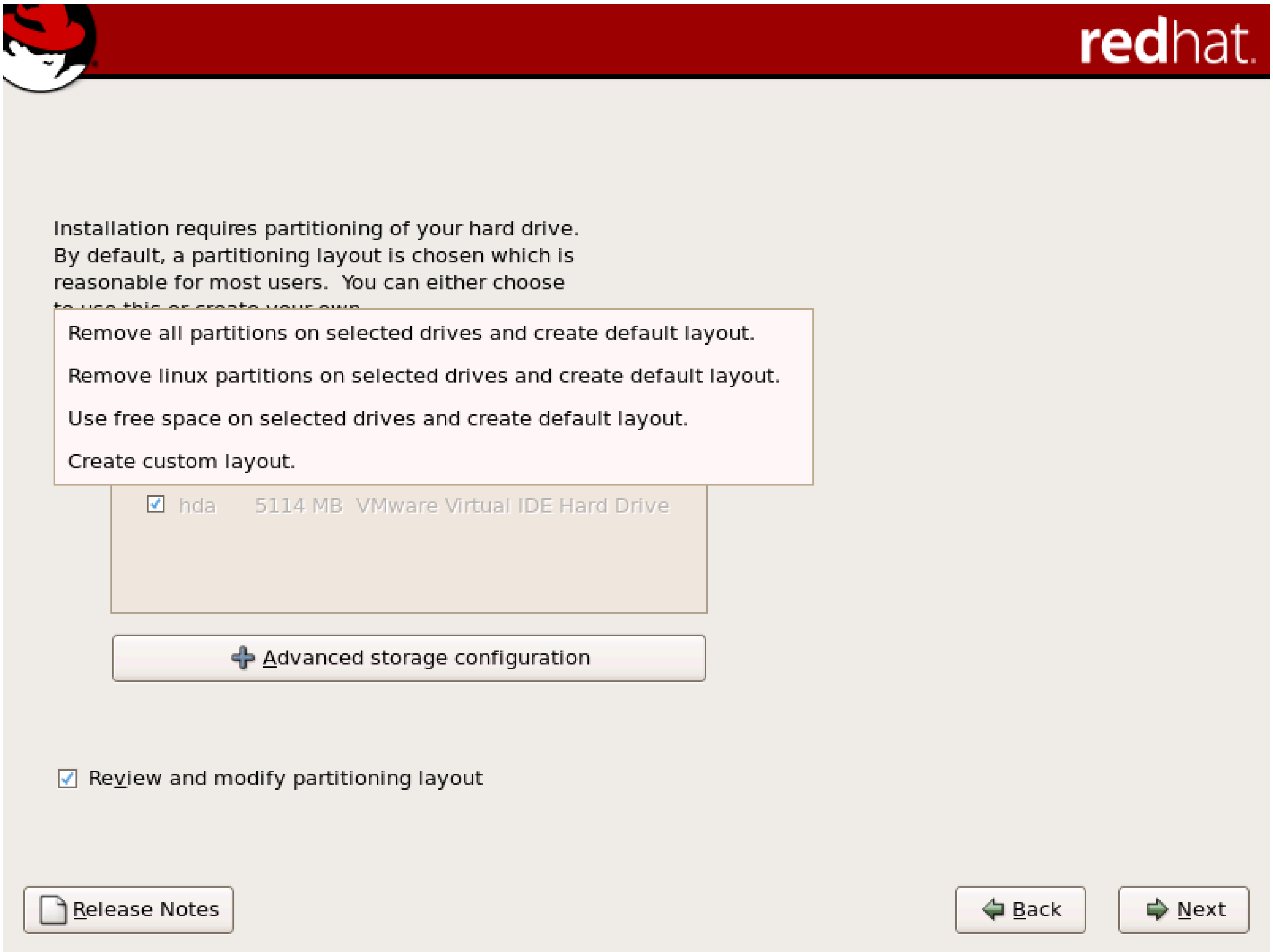

Figure 4.15. Create Default Layout

Using your mouse, choose the storage drive(s) on which you want Red Hat Enterprise Linux to be installed. If you have two or more drives, you can choose which drive(s) should contain this installation. Unselected drives, and any data on them, are not touched.

Warning

It is always a good idea to back up any data that you have on your systems. For example, if you are upgrading or creating a dual-boot system, you should back up any data you wish to keep on your drive(s). Mistakes do happen and can result in the loss of all your data.

> **Note**
>
> If you have a RAID card, be aware that some BIOSes do not support booting from the RAID card. In cases such as these, the **`/boot/`** partition must be created on a partition outside of the RAID array, such as on a separate hard drive. An internal hard drive is necessary to use for partition creation with problematic RAID cards.
>
> A **`/boot/`** partition is also necessary for software RAID setups.
>
> If you have chosen to automatically partition your system, you should select **`Review`** and manually edit your **`/boot/`** partition.

To review and make any necessary changes to the partitions created by automatic partitioning, select the **`Review`** option. After selecting **`Review`** and clicking **`Next`** to move forward, the partitions created for you in **Disk Druid** appear. You can make modifications to these partitions if they do not meet your needs.

Click **`Next`** once you have made your selections to proceed.

4.19. Partitioning Your System

If you chose one of the three automatic partitioning options and did not select **`Review`**, skip ahead to Section 4.21, “Network Configuration”.

If you chose one of the automatic partitioning options and selected **`Review`**, you can either accept the current partition settings (click **`Next`**), or modify the setup using **Disk Druid**, the manual partitioning tool.

> **Note**
>
> Please note that in the text mode installation it is not possible to work with LVM (Logical Volumes) beyond viewing the existing setup. LVM can only be set up using the graphical Disk Druid program in a graphical installation.

If you chose to create a custom layout, you must tell the installation program where to install Red Hat Enterprise Linux. This is done by defining mount points for one or more disk partitions in which Red Hat Enterprise Linux is installed. You may also need to create and/or delete partitions at this time.

> **Note**
>
> If you have not yet planned how to set up your partitions, refer to Chapter 26, *An Introduction to Disk Partitions* and Section 4.19.4, “Recommended Partitioning Scheme”. At a bare minimum, you need an appropriately-sized root partition, and a swap partition equal to twice the amount of RAM you have on the system. Itanium system users should have a **`/boot/efi/`** partition of approximately 100 MB and of type FAT (VFAT), a swap partition of at least 512 MB, and an appropriately-sized root (`/`) partition.

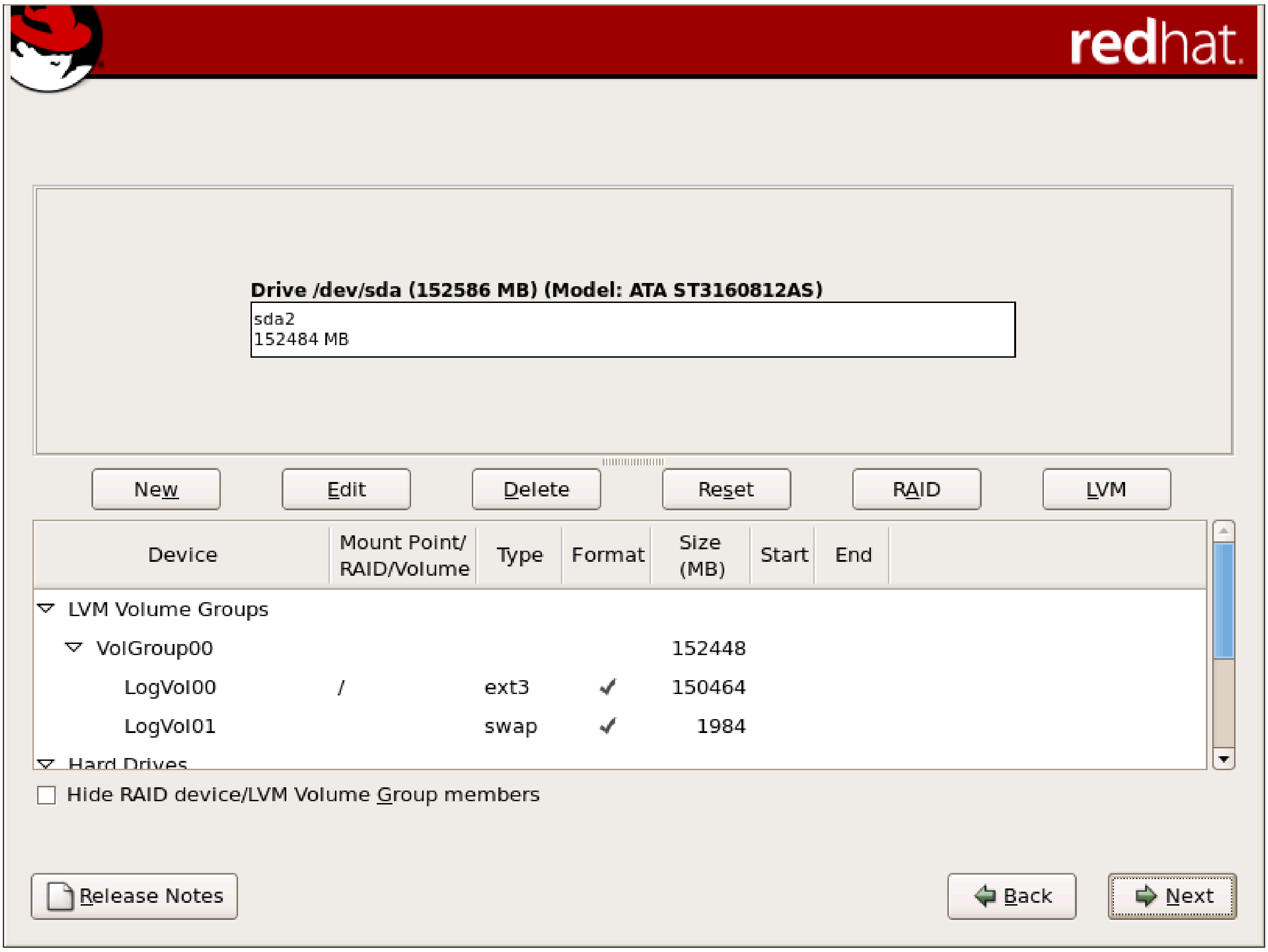

Figure 4.16. Partitioning with Disk Druid on x86, AMD64, and Intel® 64 Systems

The partitioning tool used by the installation program is **Disk Druid**. With the exception of certain esoteric situations, **Disk Druid** can handle the partitioning requirements for a typical installation.

4.19.1. Graphical Display of Hard Drive(s)

Disk Druid offers a graphical representation of your hard drive(s).

Using your mouse, click once to highlight a particular field in the graphical display. Double-click to edit an existing partition or to create a partition out of existing free space.

Above the display, you can review the name of the drive (such as /dev/hda), its size (in MB), and its model as detected by the installation program.

4.19.2. Disk Druid's Buttons

These buttons control **Disk Druid's** actions. They are used to change the attributes of a partition (for example the file system type and mount point) and also to create RAID devices. Buttons on this screen are also used to accept the changes you have made, or to exit **Disk Druid**. For further explanation, take a look at each button in order:

- **New**: Used to request a new partition. When selected, a dialog box appears containing fields (such as the mount point and size fields) that must be filled in.

- **Edit**: Used to modify attributes of the partition currently selected in the **Partitions** section. Selecting **Edit** opens a dialog box. Some or all of the fields can be edited, depending on whether the partition information has already been written to disk.

  You can also edit free space as represented in the graphical display to create a new partition within that space. Either highlight the free space and then select the **Edit** button, or double-click on the free space to edit it.

- To make a RAID device, you must first create (or reuse existing) software RAID partitions. Once you have created two or more software RAID partitions, select **Make RAID** to join the software RAID partitions into a RAID device.

- **Delete**: Used to remove the partition currently highlighted in the **Current Disk Partitions** section. You will be asked to confirm the deletion of any partition.

- **Reset**: Used to restore **Disk Druid** to its original state. All changes made will be lost if you **Reset** the partitions.

- **RAID**: Used to provide redundancy to any or all disk partitions. *It should only be used if you have experience using RAID.* To read more about RAID, see the Red Hat Enterprise Linux Deployment Guide.

  To make a RAID device, you must first create software RAID partitions. Once you have created two or more software RAID partitions, select **RAID** to join the software RAID partitions into a RAID device.

- **LVM**: Allows you to create an LVM logical volume. The role of LVM (Logical Volume Manager) is to present a simple logical view of underlying physical storage space, such as a hard drive(s). LVM manages individual physical disks — or to be more precise, the individual partitions present on them. *It should only be used if you have experience using LVM.* To read more about LVM, see the Red Hat Enterprise Linux Deployment Guide. Note, LVM is only available in the graphical installation program.

  To create an LVM logical volume, you must first create partitions of type physical volume (LVM). Once you have created one or more physical volume (LVM) partitions, select **LVM** to create an LVM logical volume.

4.19.3. Partition Fields

Above the partition hierarchy are labels which present information about the partitions you are creating. The labels are defined as follows:

- **Device**: This field displays the partition's device name.
- **Mount Point/RAID/Volume**: A mount point is the location within the directory hierarchy at which a volume exists; the volume is "mounted" at this location. This field indicates where the partition is mounted. If a partition exists, but is not set, then you need to define its mount point. Double-click on the partition or click the **Edit** button.
- **Type**: This field shows the partition's file system type (for example, ext2, ext3, or vfat).
- **Format**: This field shows if the partition being created will be formatted.
- **Size (MB)**: This field shows the partition's size (in MB).
- **Start**: This field shows the cylinder on your hard drive where the partition begins.
- **End**: This field shows the cylinder on your hard drive where the partition ends.

`Hide RAID device/LVM Volume Group members`: Select this option if you do not want to view any RAID device or LVM Volume Group members that have been created.

4.19.4. Recommended Partitioning Scheme

4.19.4.1. Itanium systems

*Unless you have a reason for doing otherwise, we recommend that you create the following partitions for **`Itanium`** systems:*

- A **`/boot/efi/`** partition (100 MB minimum) — the partition mounted on **`/boot/efi/`** contains all the installed kernels, the initrd images, and ELILO configuration files.

  Warning

  You must create a **`/boot/efi/`** partition of type VFAT and at least 100 MB in size as the first primary partition.

- A swap partition (at least 256 MB) — swap partitions are used to support virtual memory. In other words, data is written to a swap partition when there is not enough RAM to store the data your system is processing.

  In years past, the recommended amount of swap space increased linearly with the amount of RAM in the system. But because the amount of memory in modern systems has increased into the hundreds of gigabytes, it is now recognized that the amount of swap space that a system needs is a function of the memory workload running on that system. However, given that swap space is usually designated at install time, and that it can be difficult to determine beforehand the memory workload of a system, we recommend determining system swap using the following table.

  Table 4.2. Recommended System Swap Space

Amount of RAM in the System	Recommended Amount of Swap Space
4GB of RAM or less	a minimum of 2GB of swap space
4GB to 16GB of RAM	a minimum of 4GB of swap space
16GB to 64GB of RAM	a minimum of 8GB of swap space
64GB to 256GB of RAM	a minimum of 16GB of swap space
256GB to 512GB of RAM	a minimum of 32GB of swap space

  Note that you can obtain better performance by distributing swap space over multiple storage devices, particularly on systems with fast drives, controllers, and interfaces.

- A root partition (3.0 GB - 5.0 GB) — this is where "`/`" (the root directory) is located. In this setup, all files (except those stored in **`/boot/efi`**) are on the root partition.

  A 3.0 GB partition allows you to install a minimal installation, while a 5.0 GB root partition lets you perform a full installation, choosing all package groups.

4.19.4.2. x86, AMD64, and Intel® 64 systems

Unless you have a reason for doing otherwise, we recommend that you create the following partitions for x86, AMD64, and Intel® 64 systems:

- A swap partition (at least 256 MB) — swap partitions are used to support virtual memory. In other words, data is written to a swap partition when there is not enough RAM to store the data your system is processing.

  In years past, the recommended amount of swap space increased linearly with the amount of RAM in the system. But because the amount of memory in modern systems has increased into the hundreds of gigabytes, it is now recognized that the amount of swap space that a system needs is a function of the memory workload running on that system. However, given that swap space is usually designated at install time, and that it can be difficult to determine beforehand the memory workload of a system, we recommend determining system swap using the following table.

  Table 4.3. Recommended System Swap Space

Amount of RAM in the System	Recommended Amount of Swap Space
4GB of RAM or less	a minimum of 2GB of swap space
4GB to 16GB of RAM	a minimum of 4GB of swap space
16GB to 64GB of RAM	a minimum of 8GB of swap space
64GB to 256GB of RAM	a minimum of 16GB of swap space
256GB to 512GB of RAM	a minimum of 32GB of swap space

  Note that you can obtain better performance by distributing swap space over multiple storage devices, particularly on systems with fast drives, controllers, and interfaces.

- A **/boot/** partition (250 MB) — the partition mounted on **/boot/** contains the operating system kernel (which allows your system to boot Red Hat Enterprise Linux), along with files used during the bootstrap process. Due to limitations, creating a native ext3 partition to hold these files is required. For most users, a 250 MB boot partition is sufficient.

  Note

  If your hard drive is more than 1024 cylinders (and your system was manufactured more than two years ago), you may need to create a **/boot/** partition if you want the **/** (root) partition to use all of the remaining space on your hard drive.

  Note

  If you have a RAID card, be aware that some BIOSes do not support booting from the RAID card. In cases such as these, the **/boot/** partition must be created on a partition outside of the RAID array, such as on a separate hard drive.

- A **root** partition (3.0 GB - 5.0 GB) — this is where "**/**" (the root directory) is located. In this setup, all files (except those stored in **/boot**) are on the root partition.

  A 3.0 GB partition allows you to install a minimal installation, while a 5.0 GB root partition lets you perform a full installation, choosing all package groups.

- A **home** partition (at least 100 MB) — for storing user data separately from system data. This will be a dedicated partition within a volume group for the **/home** directory. This will enable you to upgrade or reinstall Red Hat Enterprise Linux without erasing user data files.

Warning — do not place /var on network storage

Red Hat Enterprise Linux 5.11 does not support having a separate **`/var`** on a network filesystem (for example, NFS, iSCSI, or NBD) The **`/var`** directory contains critical data that must be read from or written to during the boot process before establishing network services.

However, you may have **`/var/spool`**, **`/var/www`** or other subdirectories on a separate network disk, just not the complete **`/var`** filesystem.

4.19.5. Adding Partitions

To add a new partition, select the **New** button. A dialog box appears (refer to Figure 4.17, “Creating a New Partition”).

Note

You must dedicate at least one partition for this installation, and optionally more. For more information, refer to Chapter 26, *An Introduction to Disk Partitions*.

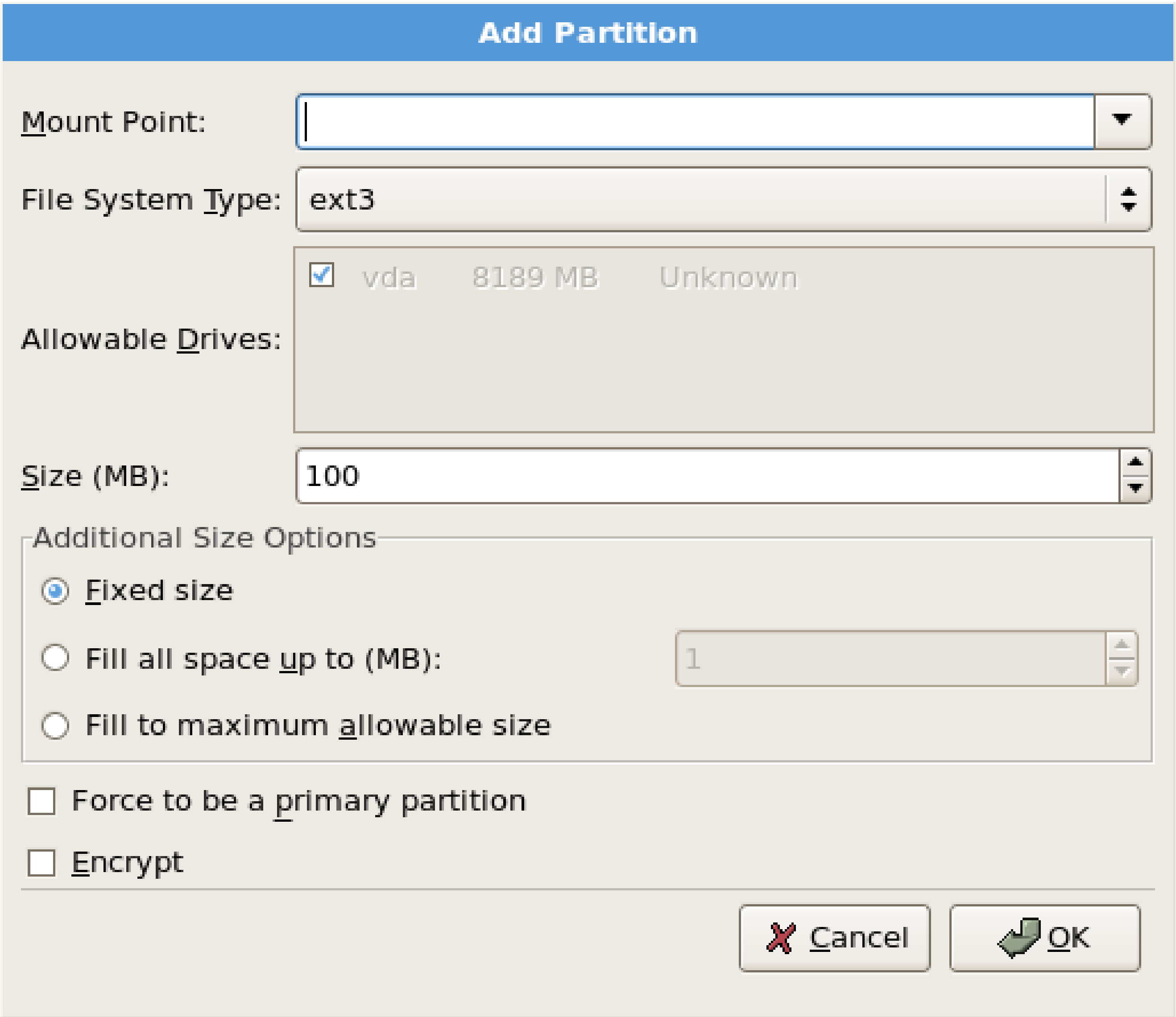

Figure 4.17. Creating a New Partition

- **`Mount Point`**: Enter the partition's mount point. For example, if this partition should be the root partition, enter `/`; enter **`/boot`** for the **`/boot`** partition, and so on. You can also use the pull-down menu to choose the correct mount point for your partition. For a swap partition the mount point should not be set - setting the filesystem type to swap is sufficient.

- **`File System Type`**: Using the pull-down menu, select the appropriate file system type for this partition. For more information on file system types, refer to Section 4.19.5.1, "File System Types".

- **`Allowable Drives`**: This field contains a list of the hard disks installed on your system. If a hard disk's box is highlighted, then a desired partition can be created on that hard disk. If the box is *not* checked, then the partition will *never* be created on that hard disk. By using different checkbox settings, you can have **Disk Druid** place partitions where you need them, or let **Disk Druid** decide where partitions should go.

- **`Size (MB)`**: Enter the size (in megabytes) of the partition. Note, this field starts with 100 MB; unless changed, only a 100 MB partition will be created.

- **`Additional Size Options`**: Choose whether to keep this partition at a fixed size, to allow it to "grow" (fill up the available hard drive space) to a certain point, or to allow it to grow to fill any remaining hard drive space available.

If you choose **Fill all space up to (MB)**, you must give size constraints in the field to the right of this option. This allows you to keep a certain amount of space free on your hard drive for future use.

- **Force to be a primary partition**: Select whether the partition you are creating should be one of the first four partitions on the hard drive. If unselected, the partition is created as a logical partition. Refer to Section 26.1.3, “Partitions within Partitions — An Overview of Extended Partitions”, for more information.

- **Encrypt**: Choose whether to encrypt the partition so that the data stored on it cannot be accessed without a passphrase, even if the storage device is connected to another system. Refer to Chapter 29, *Disk Encryption Guide* for information on encryption of storage devices. If you select this option, the installer prompts you to provide a passphrase before it writes the partition to the disk.

- **OK**: Select **OK** once you are satisfied with the settings and wish to create the partition.

- **Cancel**: Select **Cancel** if you do not want to create the partition.

4.19.5.1. File System Types

Red Hat Enterprise Linux allows you to create different partition types, based on the file system they will use. The following is a brief description of the different file systems available, and how they can be utilized.

- **ext3** — The ext3 file system is based on the ext2 file system and has one main advantage — journaling. Using a journaling file system reduces time spent recovering a file system after a crash as there is no need to **fsck** [2] the file system. A maximum file system size of 16TB is supported for ext3. The ext3 file system is selected by default and is highly recommended.

- **ext2** — An ext2 file system supports standard Unix file types (regular files, directories, symbolic links, etc). It provides the ability to assign long file names, up to 255 characters.

- **physical volume (LVM)** — Creating one or more physical volume (LVM) partitions allows you to create an LVM logical volume. LVM can improve performance when using physical disks. For more information regarding LVM, see the Red Hat Enterprise Linux Deployment Guide.

- **software RAID** — Creating two or more software RAID partitions allows you to create a RAID device. For more information regarding RAID, see chapter *RAID (Redundant Array of Independent Disks)* in the Red Hat Enterprise Linux Deployment Guide.

- **swap** — Swap partitions are used to support virtual memory. In other words, data is written to a swap partition when there is not enough RAM to store the data your system is processing. See the Red Hat Enterprise Linux Deployment Guide for additional information.

- **vfat** — The VFAT file system is a Linux file system that is compatible with Microsoft Windows long filenames on the FAT file system. This file system must be used for the **/boot/efi/** partition on Itanium systems.

4.19.6. Editing Partitions

To edit a partition, select the **Edit** button or double-click on the existing partition.

Note

If the partition already exists on your disk, you can only change the partition's mount point. To make any other changes, you must delete the partition and recreate it.

4.19.7. Deleting a Partition

To delete a partition, highlight it in the **Partitions** section and click the **Delete** button. Confirm the deletion when prompted.

For further installation instructions for x86, AMD64, and Intel® 64 systems, skip to Section 4.20, “x86, AMD64, and Intel® 64 Boot Loader Configuration”.

For further installation instructions for Itanium systems, skip to Section 4.21, “Network Configuration”.

4.20. x86, AMD64, and Intel® 64 Boot Loader Configuration

To boot the system without boot media, you usually need to install a boot loader. A boot loader is the first software program that runs when a computer starts. It is responsible for loading and transferring control to the operating system kernel software. The kernel, in turn, initializes the rest of the operating system.

GRUB (GRand Unified Bootloader), which is installed by default, is a very powerful boot loader. GRUB can load a variety of free operating systems, as well as proprietary operating systems with chain-loading (the mechanism for loading unsupported operating systems, such as DOS or Windows, by loading another boot loader).

Figure 4.18. Boot Loader Configuration

If you do not want to install GRUB as your boot loader, click **`Change boot loader`**, where you can choose not to install a boot loader at all.

If you already have a boot loader that can boot Red Hat Enterprise Linux and do not want to overwrite your current boot loader, choose **`Do not install a boot loader`** by clicking on the **`Change boot loader`** button.

Warning

If you choose not to install GRUB for any reason, you will not be able to boot the system directly, and you must use another boot method (such as a commercial boot loader application). Use this option only if you are sure you have another way of booting the system!

Every bootable partition is listed, including partitions used by other operating systems. The partition holding the system's root file system has a **`Label`** of **`Red Hat Enterprise Linux`** (for GRUB). Other partitions may also have boot labels. To add or change the boot label for other partitions that have been detected by the installation program, click once on the partition to select it. Once selected, you can change the boot label by clicking the **`Edit`** button.

Select **`Default`** beside the preferred boot partition to choose your default bootable OS. You cannot move forward in the installation unless you choose a default boot image.

Note

The **`Label`** column lists what you must enter at the boot prompt, in non-graphical boot loaders, in order to boot the desired operating system.

Once you have loaded the GRUB boot screen, use the arrow keys to choose a boot label or type **`e`** for edit. You are presented with a list of items in the configuration file for the boot label you have selected.

Boot loader passwords provide a security mechanism in an environment where physical access to your server is available.

If you are installing a boot loader, you should create a password to protect your system. Without a boot loader password, users with access to your system can pass options to the kernel which can compromise your system security. With a boot loader password in place, the password must first be entered before selecting any non-standard boot options. However, it is still possible for someone with physical access to the machine to boot from a diskette, CD-ROM, or USB media if the BIOS supports it. Security plans which include boot loader passwords should also address alternate boot methods.

If you choose to use a boot loader password to enhance your system security, be sure to select the checkbox labeled **`Use a boot loader password`**.

Once selected, enter a password and confirm it.

GRUB recognizes only the QWERTY keyboard layout

When selecting a GRUB password, be aware that GRUB recognizes only the QWERTY keyboard layout, regardless of the keyboard actually attached to the system. If you use a keyboard with a significantly different layout, it might be more effective to memorize a pattern of keystrokes rather than the word that the pattern produces.

To configure more advanced boot loader options, such as changing the drive order or passing options to the kernel, be sure **`Configure advanced boot loader options`** is selected before clicking **`Next`**.

Important — multipath devices

If you are installing Red Hat Enterprise Linux on a network device accessible through multiple paths, ensure that you select **`Configure advanced boot loader options`** before you click **`Next`**.

4.20.1. Advanced Boot Loader Configuration

Now that you have chosen which boot loader to install, you can also determine where you want the boot loader to be installed. You may install the boot loader in one of two places:

- The master boot record (MBR) — This is the recommended place to install a boot loader, unless the MBR already starts another operating system loader, such as System Commander. The MBR is a special area on your hard drive that is automatically loaded by your computer's BIOS, and is

the earliest point at which the boot loader can take control of the boot process. If you install it in the MBR, when your machine boots, GRUB presents a boot prompt. You can then boot Red Hat Enterprise Linux or any other operating system that you have configured the boot loader to boot.

- The first sector of your boot partition — This is recommended if you are already using another boot loader on your system. In this case, your other boot loader takes control first. You can then configure that boot loader to start GRUB, which then boots Red Hat Enterprise Linux.

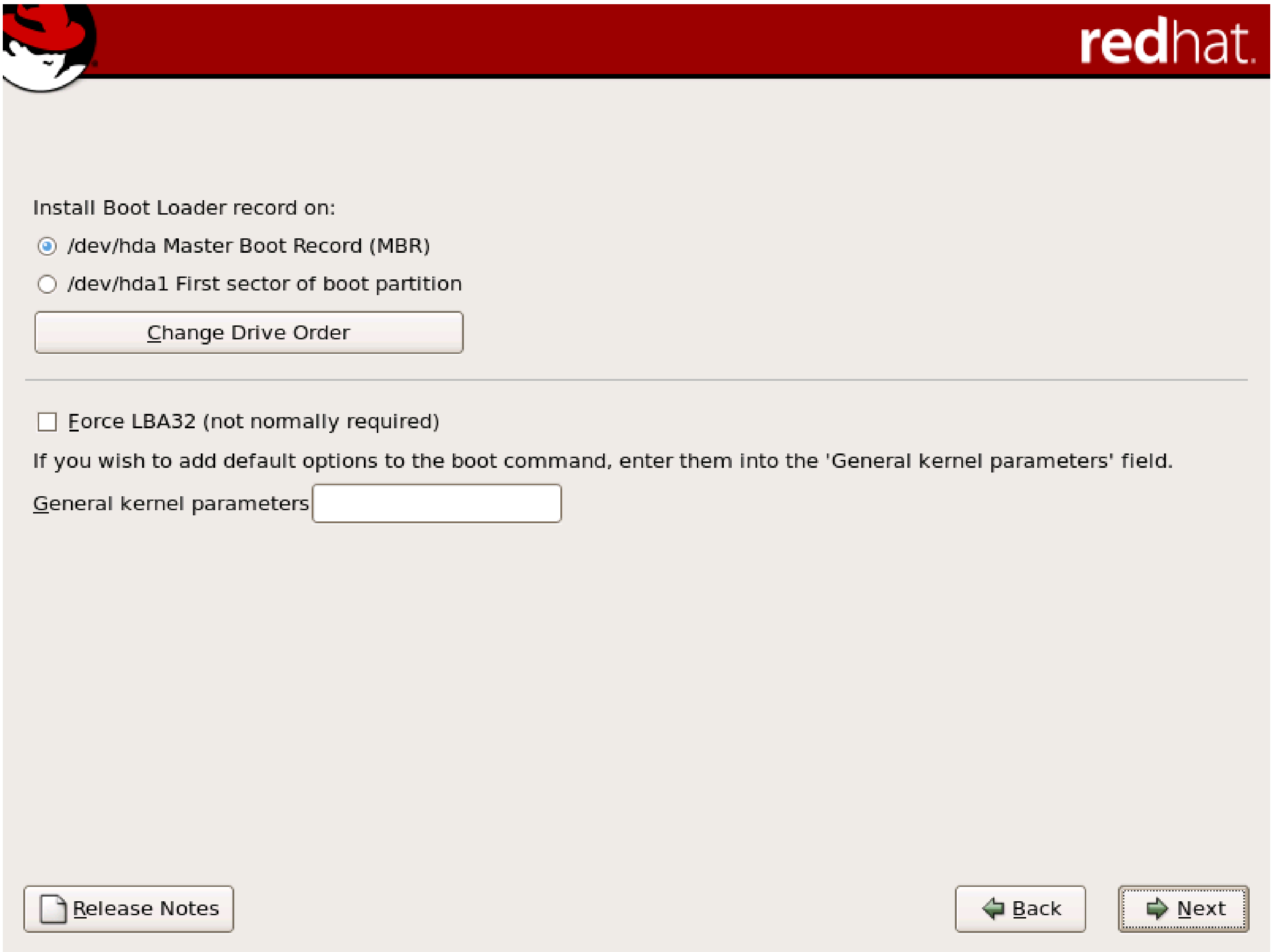

Figure 4.19. Boot Loader Installation

> **Note**
>
> If you have a RAID card, be aware that some BIOSes do not support booting from the RAID card. In cases such as these, the boot loader *should not* be installed on the MBR of the RAID array. Rather, the boot loader should be installed on the MBR of the same drive as the **`/boot/`** partition was created.

If your system only uses Red Hat Enterprise Linux, you should choose the MBR.

Click the **`Change Drive Order`** button if you would like to rearrange the drive order or if your BIOS does not return the correct drive order. Changing the drive order may be useful if you have multiple SCSI adapters, or both SCSI and IDE adapters, and you want to boot from the SCSI device.

The **Force LBA32 (not normally required)** option allows you to exceed the 1024 cylinder limit for the **/boot/** partition. If you have a system which supports the LBA32 extension for booting operating systems above the 1024 cylinder limit, and you want to place your **/boot/** partition above cylinder 1024, you should select this option.

Note

While partitioning your hard drive, keep in mind that the BIOS in some older systems cannot access more than the first 1024 cylinders on a hard drive. If this is the case, leave enough room for the **/boot** Linux partition on the first 1024 cylinders of your hard drive to boot Linux. The other Linux partitions can be after cylinder 1024.

In **parted**, 1024 cylinders equals 528MB. For more information, refer to:

```
http://www.pcguide.com/ref/hdd/bios/sizeMB504-c.html
```

To add default options to the boot command, enter them into the **Kernel parameters** field. Any options you enter are passed to the Linux kernel every time it boots.

Important — multipath devices

If you are installing Red Hat Enterprise Linux on a network device accessible through multiple paths, ensure that GRUB is installed on the MBR of **/dev/mapper/mpath0**.

4.20.2. Rescue Mode

Rescue mode provides the ability to boot a small Red Hat Enterprise Linux environment entirely from boot media or some other boot method instead of the system's hard drive. There may be times when you are unable to get Red Hat Enterprise Linux running completely enough to access files on your system's hard drive. Using rescue mode, you can access the files stored on your system's hard drive, even if you cannot actually run Red Hat Enterprise Linux from that hard drive. If you need to use rescue mode, try the following method:

- Using the CD-ROM to boot an x86, AMD64, or Intel® 64 system, type **linux rescue** at the installation boot prompt. Itanium users should type **elilo linux rescue** to enter rescue mode.

For additional information, see the Red Hat Enterprise Linux Deployment Guide.

4.20.3. Alternative Boot Loaders

If you do not wish to use a boot loader, you have several alternatives:

LOADLIN

You can load Linux from MS-DOS. Unfortunately, this requires a copy of the Linux kernel (and an initial RAM disk, if you have a SCSI adapter) to be available on an MS-DOS partition. The only way to accomplish this is to boot your Red Hat Enterprise Linux system using some other method (for example, from a boot CD-ROM) and then copy the kernel to an MS-DOS partition. LOADLIN is available from

ftp://metalab.unc.edu/pub/Linux/system/boot/dualboot/

and associated mirror sites.

SYSLINUX

SYSLINUX is an MS-DOS program very similar to LOADLIN. It is also available from

ftp://metalab.unc.edu/pub/Linux/system/boot/loaders/

and associated mirror sites.

Commercial boot loaders

You can load Linux using commercial boot loaders. For example, System Commander and Partition Magic are able to boot Linux (but still require GRUB to be installed in your Linux root partition).

> **Note**
>
> Boot loaders such as LOADLIN and System Commander are considered to be third-party boot loaders and are not supported by Red Hat.

4.20.4. SMP Motherboards and GRUB

In previous versions of Red Hat Enterprise Linux there were two different kernel versions, a uniprocessor version and an SMP version. In Red Hat Enterprise Linux 5.11 the kernel is SMP-enabled by default and will take advantage of multiple core, hyperthreading, and multiple CPU capabilities when they are present. This same kernel can run on single CPUs with a single core and no hyperthreading.

4.21. Network Configuration

If you do not have a network device, this screen does not appear during your installation and you should advance to Section 4.22, “Time Zone Configuration”.

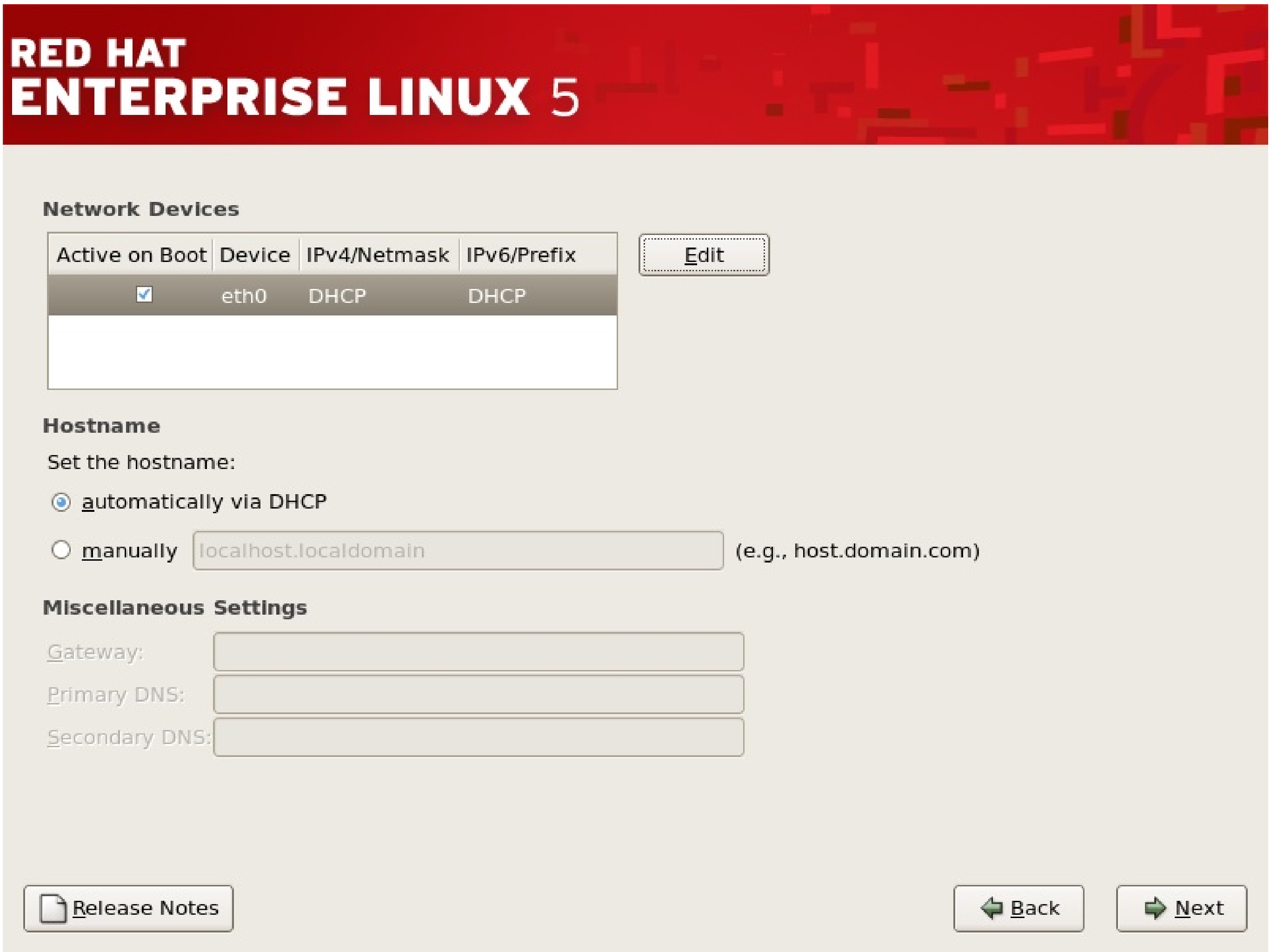

Figure 4.20. Network Configuration

The installation program automatically detects any network devices you have and displays them in the **`Network Devices`** list.

When you have selected a network device, click **`Edit`**. From the **`Edit Interface`** dialog, you can choose to configure the IP address and Netmask (for IPv4 - Prefix for IPv6) of the device to use DHCP or to use static settings. If you do not have DHCP client access or you are unsure what to provide here, contact your network administrator.

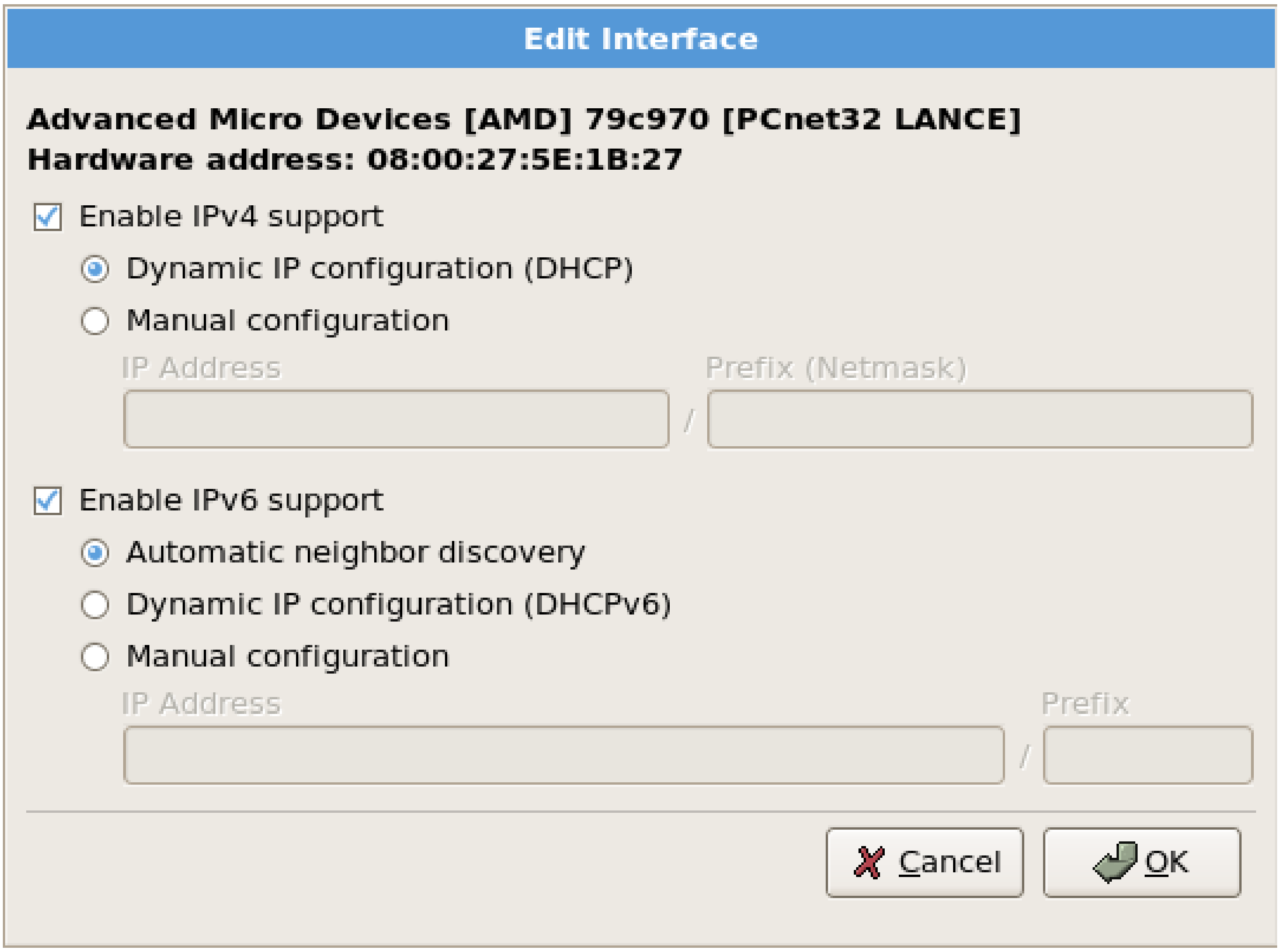

Figure 4.21. Editing a Network Device

> **Note**
>
> Do not use the numbers as seen in this sample configuration. These values will not work for your own network configuration. If you are not sure what values to enter, contact your network administrator for assistance.

If you have a hostname (fully qualified domain name) for the network device, you can choose to have DHCP (Dynamic Host Configuration Protocol) automatically detect it or you can manually enter the hostname in the field provided.

Finally, if you entered the IP and Netmask information manually, you may also enter the Gateway address and the Primary and Secondary DNS addresses.

> **Note**
>
> Even if your computer is not part of a network, you can enter a hostname for your system. If you do not take this opportunity to enter a name, your system will be known as **`localhost`**.

> **Note**
>
> To change your network configuration after you have completed the installation, use the **Network Administration Tool**.
>
> Type the **`system-config-network`** command in a shell prompt to launch the **Network Administration Tool**. If you are not root, it prompts you for the root password to continue.

4.22. Time Zone Configuration

Set your time zone by selecting the city closest to your computer's physical location. Click on the map to zoom in to a particular geographical region of the world.

From here there are two ways for you to select your time zone:

- Using your mouse, click on the interactive map to select a specific city (represented by a yellow dot). A red **X** appears indicating your selection.
- You can also scroll through the list at the bottom of the screen to select your time zone. Using your mouse, click on a location to highlight your selection.

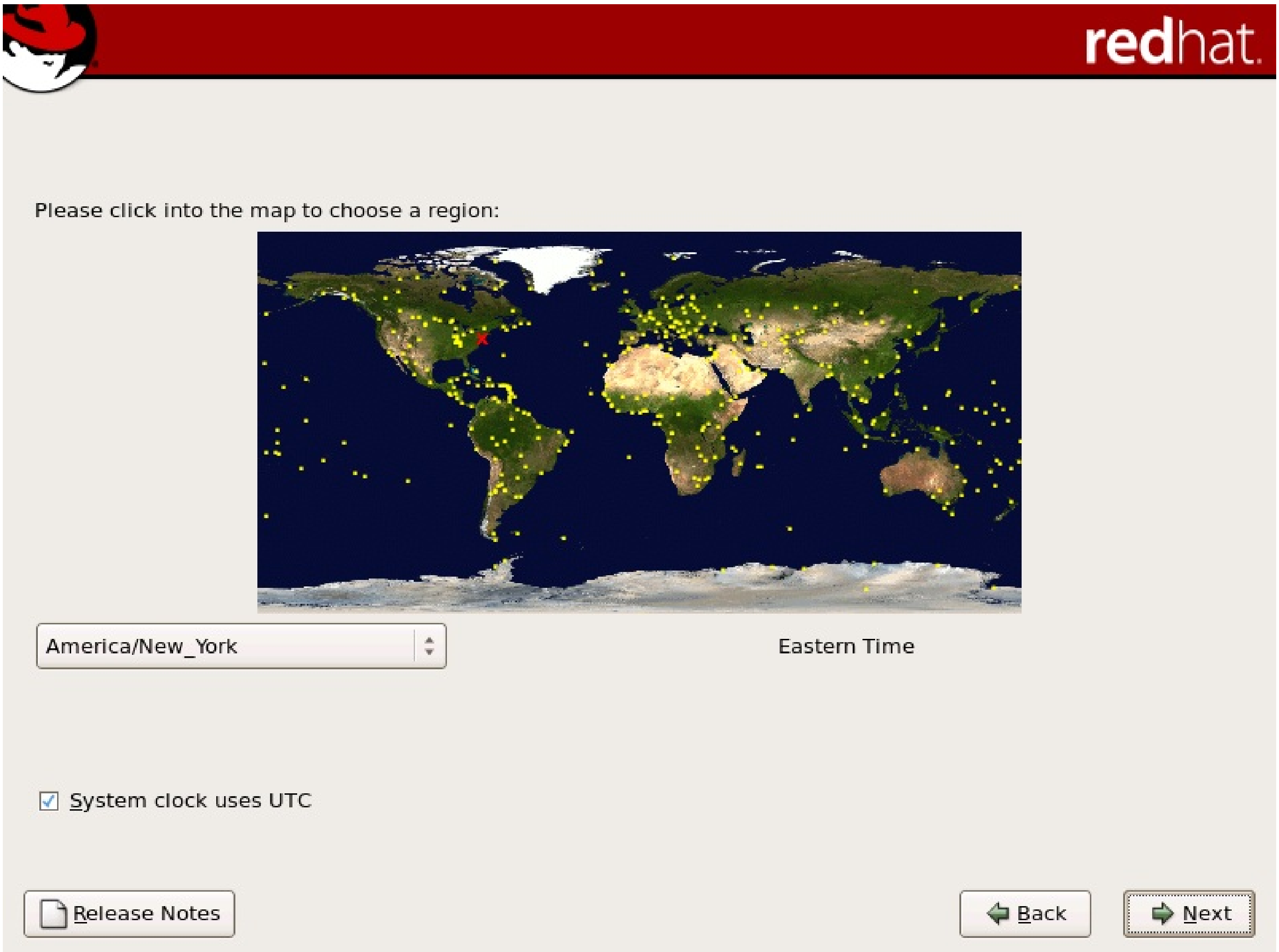

Figure 4.22. Configuring the Time Zone

Select **`System Clock uses UTC`** if you know that your system is set to UTC.

> **Note**
>
> To change your time zone configuration after you have completed the installation, use the **Time and Date Properties Tool**.
>
> Type the **`system-config-date`** command in a shell prompt to launch the **Time and Date Properties Tool**. If you are not root, it prompts you for the root password to continue.
>
> To run the **Time and Date Properties Tool** as a text-based application, use the command **`timeconfig`**.

4.23. Set Root Password

Setting up a root account and password is one of the most important steps during your installation. Your root account is similar to the administrator account used on Windows NT machines. The root account is used to install packages, upgrade RPMs, and perform most system maintenance. Logging in as root gives you complete control over your system.

> **Note**
>
> The root user (also known as the superuser) has complete access to the entire system; for this reason, logging in as the root user is best done *only* to perform system maintenance or administration.

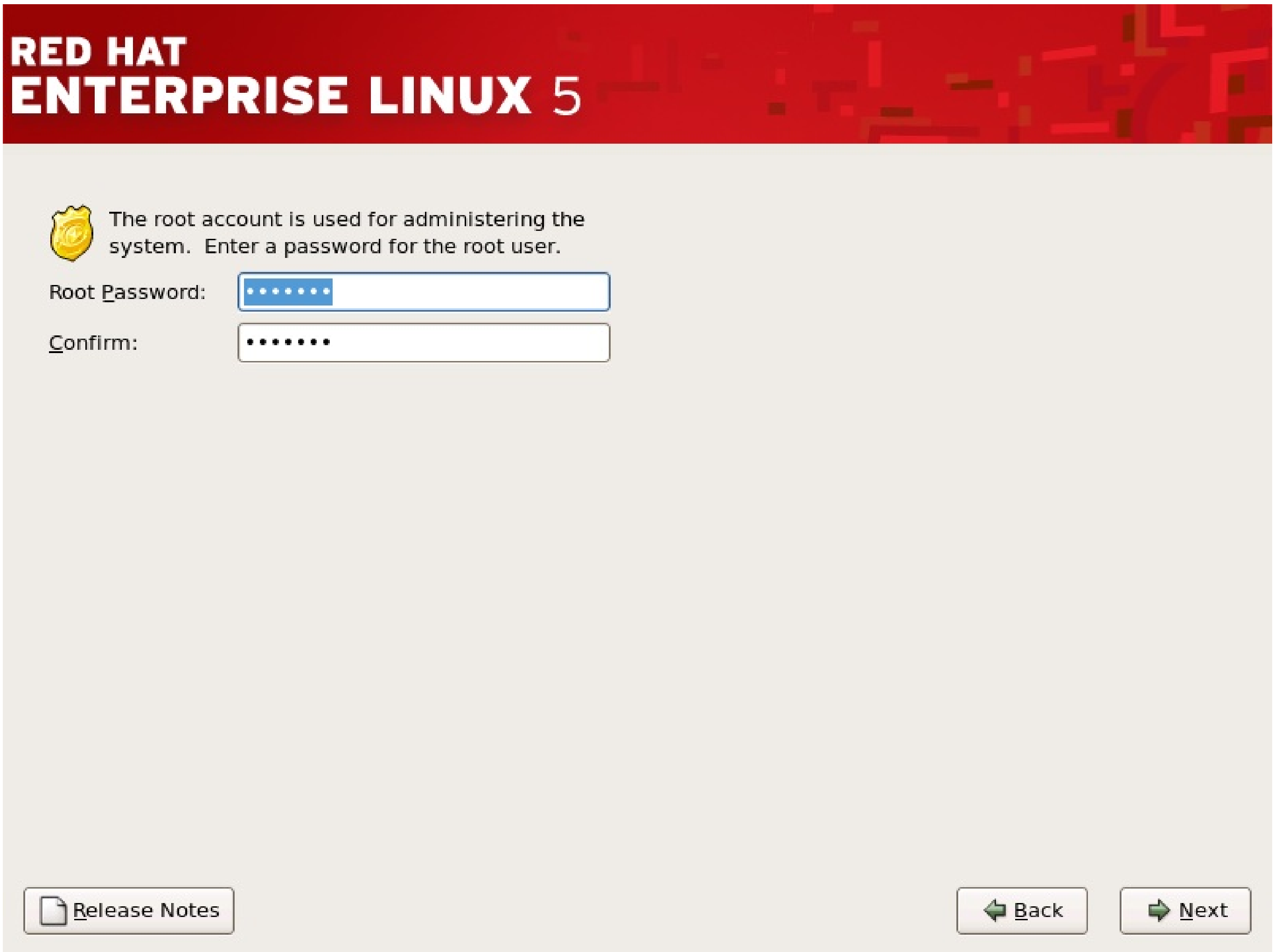

Figure 4.23. Root Password

Use the root account only for system administration. Create a non-root account for your general use and **su -** to root when you need to fix something quickly. These basic rules minimize the chances of a typo or an incorrect command doing damage to your system.

Note

To become root, type **su -** at the shell prompt in a terminal window and then press **Enter**. Then, enter the root password and press **Enter**.

The installation program prompts you to set a root password [3] for your system. *You cannot proceed to the next stage of the installation process without entering a root password.*

The root password must be at least six characters long; the password you type is not echoed to the screen. You must enter the password twice; if the two passwords do not match, the installation program asks you to enter them again.

You should make the root password something you can remember, but not something that is easy for someone else to guess. Your name, your phone number, *qwerty*, *password, root*, *123456*, and *anteater* are all examples of bad passwords. Good passwords mix numerals with upper and lower case letters and do not contain dictionary words: *Aard387vark* or *420BMttNT*, for example. Remember that the password is case-sensitive. If you write down your password, keep it in a secure place. However, it is recommended that you do not write down this or any password you create.

> **Note**
>
> Do not use one of the example passwords offered in this manual. Using one of these passwords could be considered a security risk.

> **Note**
>
> To change your root password after you have completed the installation, use the **Root Password Tool**.
>
> Type the **`system-config-rootpassword`** command in a shell prompt to launch the **Root Password Tool**. If you are not root, it prompts you for the root password to continue.

4.24. Package Group Selection

Now that you have made most of the choices for your installation, you are ready to confirm the default package selection or customize packages for your system.

The **`Package Installation Defaults`** screen appears and details the default package set for your Red Hat Enterprise Linux installation. This screen varies depending on the version of Red Hat Enterprise Linux you are installing.

If you choose to accept the current package list, skip ahead to Section 4.25, “Preparing to Install”.

To customize your package set further, select the **`Customize now`** option on the screen. Clicking **`Next`** takes you to the **`Package Group Selection`** screen.

You can select package groups, which group components together according to function (for example, **`X Window System`** and **`Editors`**), individual packages, or a combination of the two.

> **Note**
>
> Users of Itanium systems who want support for developing or running 32-bit applications are encouraged to select the **`Compatibility Arch Support`** and **`Compatibility Arch Development Support`** packages to install architecture specific support for their systems.

To select a component, click on the checkbox beside it (refer to Figure 4.24, “Package Group Selection”).

Figure 4.24. Package Group Selection

Select each component you wish to install.

Once a package group has been selected, if optional components are available you can click on **`Optional packages`** to view which packages are installed by default, and to add or remove optional packages from that group. If there are no optional components this button will be disabled.

Figure 4.25. Package Group Details

4.25. Preparing to Install

4.25.1. Prepare to Install

A screen preparing you for the installation of Red Hat Enterprise Linux now appears.

For your reference, a complete log of your installation can be found in **`/root/install.log`** once you reboot your system.

> **Warning**
>
> If, for some reason, you would rather not continue with the installation process, this is your last opportunity to safely cancel the process and reboot your machine. Once you press the **`Next`** button, partitions are written and packages are installed. If you wish to abort the installation, you should reboot now before any existing information on any hard drive is rewritten.

To cancel this installation process, press your computer's Reset button or use the **`Control`**+**`Alt`**+**`Delete`** key combination to restart your machine.

4.26. Installing Packages

At this point there is nothing left for you to do until all the packages have been installed. How quickly this happens depends on the number of packages you have selected and your computer's speed.

4.27. Installation Complete

Congratulations! Your Red Hat Enterprise Linux installation is now complete!

The installation program prompts you to prepare your system for reboot. Remember to remove any installation media if it is not ejected automatically upon reboot.

After your computer's normal power-up sequence has completed, the graphical boot loader prompt appears at which you can do any of the following things:

- Press **Enter** — causes the default boot entry to be booted.
- Select a boot label, followed by **Enter** — causes the boot loader to boot the operating system corresponding to the boot label.
- Do nothing — after the boot loader's timeout period, (by default, five seconds) the boot loader automatically boots the default boot entry.

Do whatever is appropriate to boot Red Hat Enterprise Linux. One or more screens of messages should scroll by. Eventually, a **login:** prompt or a GUI login screen (if you installed the X Window System and chose to start X automatically) appears.

The first time you start your Red Hat Enterprise Linux system in run level 5 (the graphical run level), the **Setup Agent** is presented, which guides you through the Red Hat Enterprise Linux configuration. Using this tool, you can set your system time and date, install software, register your machine with Red Hat Network, and more. The **Setup Agent** lets you configure your environment at the beginning, so that you can get started using your Red Hat Enterprise Linux system quickly.

For information on registering your Red Hat Enterprise Linux subscription, refer to Chapter 25, *Register Your System and Apply Subscriptions*.

4.28. Itanium Systems — Booting Your Machine and Post-Installation Setup

This section describes how to boot your Itanium into Red Hat Enterprise Linux and how to set your EFI console variables so that Red Hat Enterprise Linux is automatically booted when the machine is powered on.

After you reboot your system at the end of the installation program, type the following command to boot into Red Hat Enterprise Linux:

```
elilo
```

After you type **elilo**, the default kernel listed in the **/boot/efi/elilo.conf** configuration file is loaded. (The first kernel listed in the file is the default.)

If you want to load a different kernel, type the label name of the kernel from the file **/boot/efi/elilo.conf** after the **elilo** command. For example, to load the kernel named **linux**, type:

```
elilo linux
```

If you do not know the names of the installed kernels, you can view the **/boot/efi/elilo.conf** file in EFI with the following instructions:

1. At the **Shell>** prompt, change devices to the system partition (mounted as **/boot/efi** in Linux). For example, if **fs0** is the system boot partition, type **fs0:** at the EFI Shell prompt.

2. Type **ls** at the **fs0:\>** to make sure you are in the correct partition.

3. Then type:

   ```
   Shell>type elilo.conf
   ```

   This command displays the contents of the configuration file. Each stanza contains a line beginning with **label** followed by a label name for that kernel. The label name is what you type after **elilo** to boot the different kernels.

4.28.1. Post-Installation Boot Loader Options

In addition to specifying a kernel to load, you can also enter other boot options such as **single** for single user mode or **mem=1024M** to force Red Hat Enterprise Linux to use 1024 MB of memory. To pass options to the boot loader, enter the following at the EFI Shell prompt (replace ***linux*** with the label name of the kernel you want to boot and ***option*** with the boot options you want to pass to the kernel):

```
elilo linux option
```

4.28.2. Booting Red Hat Enterprise Linux Automatically

After installing Red Hat Enterprise Linux you can type **elilo** and any boot options at the EFI Shell prompt each time you wish to boot your Itanium system. However, if you wish to configure your system to boot into Red Hat Enterprise Linux automatically, you need to configure the **EFI Boot Manager**.

To configure the **EFI Boot Manager** (may vary slightly depending on your hardware):

1. Boot the Itanium system and choose **Boot option maintenance menu** from the **EFI Boot Manager** menu.

2. Choose **Add a Boot Option** from the Main Menu.

3. Select the system partition that is mounted as **/boot/efi/** in Linux.

4. Select the **elilo.efi** file.

5. At the **Enter New Description:** prompt, type **Red Hat Enterprise Linux 5**, or any name that you want to appear on the **EFI Boot Manager** menu.

6. At the **Enter Boot Option Data Type:** prompt, enter **N** for **No Boot Option** if you do not want to pass options to the ELILO boot loader. This option works for most cases. If you want to pass options to the boot loader, you can configure it in the **/boot/efi/elilo.conf** configuration file instead.

7. Answer **Yes** to the **Save changes to NVRAM** prompt. This returns you to the **EFI Boot Maintenance Manager** menu.

8. Next, you want to make the **Red Hat Enterprise Linux 5** menu item the default. A list of boot

options appears. Move the **Red Hat Enterprise Linux 5** menu item up to the top of the list by selecting it with the arrow keys and pressing the **u** key to move it up the list. You can move items down the list by selecting it and pressing the **d** key. After changing the boot order, choose **Save changes to NVRAM**. Choose **Exit** to return to the Main Menu.

9. Optionally, you can change the boot timeout value by choosing **Set Auto Boot TimeOut** => **Set Timeout Value** from the Main Menu.

10. Return to the **EFI Boot Manager** by selecting **Exit**.

4.28.2.1. Using a Startup Script

It is recommended that you configure the ELILO Boot Manager to boot Red Hat Enterprise Linux automatically. However, if you require additional commands to be executed before starting the ELILO boot loader, you can create a startup script named **startup.nsh**. The last command should be **elilo** to boot into Linux.

The **startup.nsh** script should be in the **/boot/efi** partition (**/boot/efi/startup.nsh**) and contain the following text:

```
echo -off your set of commands elilo
```

If you want to pass options to the boot loader (refer to Section 4.28.1, “Post-Installation Boot Loader Options”) add them after **elilo**.

You can either create this file after booting into Red Hat Enterprise Linux or use the editor built into the EFI shell. To use the EFI shell, at the **Shell>** prompt, change devices to the system partition (mounted as **/boot/efi** in Linux). For example, if **fs0** is the system boot partition, type **fs0:** at the EFI Shell prompt. Type **ls** to make sure you are in the correct partition. Then type **edit startup.nsh**. Type the contents of the file and save it.

The next time the system boots, EFI detects the **startup.nsh** file and use it to boot the system. To stop EFI from loading the file, type **Ctrl**+**c** . This aborts the process, and returns you to the EFI shell prompt.

---

[2] The **fsck** application is used to check the file system for metadata consistency and optionally repair one or more Linux file systems.

[3] A root password is the administrative password for your Red Hat Enterprise Linux system. You should only log in as root when needed for system maintenance. The root account does not operate within the restrictions placed on normal user accounts, so changes made as root can have implications for your entire system.

Chapter 5. Removing Red Hat Enterprise Linux

To uninstall Red Hat Enterprise Linux from your x86-based system, you must remove the Red Hat Enterprise Linux boot loader information from your master boot record (MBR).

> **Note**
>
> It is always a good idea to backup any data that you have on your system(s). Mistakes do happen and can result in the loss all of your data.

In DOS and Windows, use the Windows **fdisk** utility to create a new MBR with the *undocumented* flag **/mbr**. This *ONLY* rewrites the MBR to boot the primary DOS partition. The command should look like the following:

```
fdisk /mbr
```

If you need to remove Linux from a hard drive and have attempted to do this with the default DOS (Windows) **fdisk**, you will experience the *Partitions exist but they do not exist* problem. The best way to remove non-DOS partitions is with a tool that understands partitions other than DOS.

To begin, insert the Red Hat Enterprise Linux CD #1 and boot your system. Once you have booted off the CD, a boot prompt appears. At the boot prompt, type: **linux rescue**. This starts the rescue mode program.

You are prompted for your keyboard and language requirements. Enter these values as you would during the installation of Red Hat Enterprise Linux.

Next, a screen appears telling you that the program attempts to find a Red Hat Enterprise Linux install to rescue. Select **Skip** on this screen.

After selecting **Skip**, you are given a command prompt where you can access the partitions you would like to remove.

First, type the command **list-harddrives**. This command lists all hard drives on your system that are recognizable by the installation program, as well as their sizes in megabytes.

> **Warning**
>
> Be careful to remove only the necessary Red Hat Enterprise Linux partitions. Removing other partitions could result in data loss or a corrupted system environment.

To remove partitions, use the partitioning utility **parted**. Start **parted**, where */dev/hda* is the device on which to remove the partition:

```
parted /dev/hda
```

Using the **print** command, view the current partition table to determine the minor number of the partition to remove:

```
print
```

The **print** command also displays the partition's type (such as linux-swap, ext2, ext3, and so on). Knowing the type of the partition helps you in determining whether to remove the partition.

Remove the partition with the command **rm**. For example, to remove the partition with minor number 3:

```
rm 3
```

Important

The changes start taking place as soon as you press [Enter], so review the command before committing to it.

After removing the partition, use the **print** command to confirm that it is removed from the partition table.

Once you have removed the Linux partitions and made all of the changes you need to make, type **quit** to quit **parted**.

After quitting **parted**, type **exit** at the boot prompt to exit rescue mode and reboot your system, instead of continuing with the installation. The system should reboot automatically. If it does not, you can reboot your computer using **Control**+**Alt**+**Delete** .

Chapter 6. Troubleshooting Installation on an Intel® or AMD System

This appendix discusses some common installation problems and their solutions.

6.1. You are Unable to Boot Red Hat Enterprise Linux

6.1.1. Are You Unable to Boot With Your RAID Card?

If you have performed an installation and cannot boot your system properly, you may need to reinstall and create your partitions differently.

Some BIOSes do not support booting from RAID cards. At the end of an installation, a text-based screen showing the boot loader prompt (for example, **`GRUB:`**) and a flashing cursor may be all that appears. If this is the case, you must repartition your system.

Whether you choose automatic or manual partitioning, you must install your **`/boot`** partition outside of the RAID array, such as on a separate hard drive. An internal hard drive is necessary to use for partition creation with problematic RAID cards.

You must also install your preferred boot loader (GRUB or LILO) on the MBR of a drive that is outside of the RAID array. This should be the same drive that hosts the **`/boot/`** partition.

Once these changes have been made, you should be able to finish your installation and boot the system properly.

6.1.2. Is Your System Displaying Signal 11 Errors?

A signal 11 error, commonly know as a *segmentation fault*, means that the program accessed a memory location that was not assigned to it. A signal 11 error may be due to a bug in one of the software programs that is installed, or faulty hardware.

If you receive a fatal signal 11 error during your installation, it is probably due to a hardware error in memory on your system's bus. Like other operating systems, Red Hat Enterprise Linux places its own demands on your system's hardware. Some of this hardware may not be able to meet those demands, even if they work properly under another OS.

Ensure that you have the latest installation updates and images from Red Hat. Review the online errata to see if newer versions are available. If the latest images still fail, it may be due to a problem with your hardware. Commonly, these errors are in your memory or CPU-cache. A possible solution for this error is turning off the CPU-cache in the BIOS, if your system supports this. You could also try to swap your memory around in the motherboard slots to check if the problem is either slot or memory related.

Another option is to perform a media check on your installation CD-ROMs. The Red Hat Enterprise Linux installation program has the ability to test the integrity of the installation media. It works with the CD, DVD, hard drive ISO, and NFS ISO installation methods. Red Hat recommends that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned CDs). To use this test, type the following command at the **`boot:`** or **`yaboot:`** prompt (prepend with **`elilo`** for Itanium systems):

```
linux mediacheck
```

For more information concerning signal 11 errors, refer to:

```
http://www.bitwizard.nl/sig11/
```

6.2. Trouble Beginning the Installation

6.2.1. Problems with Booting into the Graphical Installation

There are some video cards that have trouble booting into the graphical installation program. If the installation program does not run using its default settings, it tries to run in a lower resolution mode. If that still fails, the installation program attempts to run in text mode.

One possible solution is to try using the **`resolution=`** boot option. This option may be most helpful for laptop users. Another solution to try is the **`driver=`** option to specify the driver that should be loaded for your video card. If this works, it should be reported as a bug as the installer has failed to autodetect your videocard. Refer to Chapter 8, *Additional Boot Options for Intel® and AMD Systems* for more information on boot options.

> **Note**
>
> To disable frame buffer support and allow the installation program to run in text mode, try using the **`nofb`** boot option. This command may be necessary for accessibility with some screen reading hardware.

6.3. Trouble During the Installation

6.3.1. No devices found to install Red Hat Enterprise Linux Error Message

If you receive an error message stating **`No devices found to install Red Hat Enterprise Linux`**, there is probably a SCSI controller that is not being recognized by the installation program.

Check your hardware vendor's website to determine if a driver diskette image is available that fixes your problem. For more general information on driver diskettes, refer to Chapter 7, *Updating drivers during installation on Intel and AMD systems*.

You can also refer to the *Red Hat Hardware Compatibility List*, available online at:

http://hardware.redhat.com/hcl/

6.3.2. Saving Traceback Messages Without a Diskette Drive

If you receive a traceback error message during installation, you can usually save it to a diskette.

If you do not have a diskette drive available in your system, you can **`scp`** the error message to a remote system.

When the traceback dialog appears, the traceback error message is automatically written to a file named **`/tmp/anacdump.txt`**. Once the dialog appears, switch over to a new tty (virtual console) by pressing the keys **`Ctrl+Alt+F2`** and **`scp`** the message written to **`/tmp/anacdump.txt`** to a known working remote system.

6.3.3. Trouble with Partition Tables

If you receive an error after the **Disk Partitioning Setup** (Section 4.16, “Disk Partitioning Setup”) phase of the installation saying something similar to

> The partition table on device hda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive.

you may not have a partition table on that drive or the partition table on the drive may not be recognizable by the partitioning software used in the installation program.

Users who have used programs such as **EZ-BIOS** have experienced similar problems, causing data to be lost (assuming the data was not backed up before the installation began) that could not be recovered.

No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

6.3.4. Using Remaining Space

You have a **swap** and a **/** (root) partition created, and you have selected the root partition to use the remaining space, but it does not fill the hard drive.

If your hard drive is more than 1024 cylinders, you must create a **/boot** partition if you want the **/** (root) partition to use all of the remaining space on your hard drive.

6.3.5. Other Partitioning Problems

If you are using **Disk Druid** to create partitions, but cannot move to the next screen, you probably have not created all the partitions necessary for **Disk Druid**'s dependencies to be satisfied.

You must have the following partitions as a bare minimum:

- A **/** (root) partition
- A <swap> partition of type swap

> **Note**
>
> When defining a partition's type as swap, do not assign it a mount point. **Disk Druid** automatically assigns the mount point for you.

6.3.6. Other Partitioning Problems for Itanium System Users

If you are using **Disk Druid** to create partitions, but cannot move to the next screen, you probably have not created all the partitions necessary for **Disk Druid**'s dependencies to be satisfied.

You must have the following partitions as a bare minimum:

- A **/boot/efi/** partition of type VFAT
- A **/** (root) partition
- A <swap> partition of type swap

Note

When defining a partition's type as swap, you do not have to assign it a mount point. **Disk Druid** automatically assigns the mount point for you.

6.3.7. Are You Seeing Python Errors?

During some upgrades or installations of Red Hat Enterprise Linux, the installation program (also known as **anaconda**) may fail with a Python or traceback error. This error may occur after the selection of individual packages or while trying to save the upgrade log in the **/tmp/**directory. The error may look similar to:

```
Traceback (innermost last):
File "/var/tmp/anaconda-7.1//usr/lib/anaconda/iw/progress_gui.py", line 20, in run
rc = self.todo.doInstall ()
File "/var/tmp/anaconda-7.1//usr/lib/anaconda/todo.py", line 1468, in doInstall
self.fstab.savePartitions ()
File "fstab.py", line 221, in savePartitions
sys.exit(0)
SystemExit: 0
Local variables in innermost frame:
self: <fstab.GuiFstab instance at 8446fe0>
sys: <module 'sys' (built-in)>
ToDo object:  (itodo  ToDo  p1  (dp2  S'method'  p3  (iimage  CdromInstallMethod
p4  (dp5  S'progressWindow'  p6   <failed>
```

This error occurs in some systems where links to **/tmp/** are symbolic to other locations or have been changed since creation. These symbolic or changed links are invalid during the installation process, so the installation program cannot write information and fails.

If you experience such an error, first try to download any available errata for **anaconda**. Errata can be found at:

```
http://www.redhat.com/support/errata/
```

The **anaconda** website may also be a useful reference and can be found online at:

```
http://fedoraproject.org/wiki/Anaconda
```

You can also search for bug reports related to this problem. To search Red Hat's bug tracking system, go to:

```
http://bugzilla.redhat.com/bugzilla/
```

Finally, if you are still facing problems related to this error, register your product and contact our support team. To register your product, go to:

```
http://www.redhat.com/apps/activate/
```

6.4. Problems After Installation

6.4.1. Trouble With the Graphical GRUB Screen on an x86-based System?

If you are experiencing problems with GRUB, you may need to disable the graphical boot screen. To do this, become the root user and edit the **/boot/grub/grub.conf** file.

Within the **grub.conf** file, comment out the line which begins with **splashimage** by inserting the **#** character at the beginning of the line.

Press **Enter** to exit the editing mode.

Once the boot loader screen has returned, type **b** to boot the system.

Once you reboot, the **grub.conf** file is reread and any changes you have made take effect.

You may re-enable the graphical boot screen by uncommenting (or adding) the above line back into the **grub.conf** file.

6.4.2. Booting into a Graphical Environment

If you have installed the X Window System but are not seeing a graphical desktop environment once you log into your Red Hat Enterprise Linux system, you can start the X Window System graphical interface using the command **startx**.

Once you enter this command and press **Enter**, the graphical desktop environment is displayed.

Note, however, that this is just a one-time fix and does not change the log in process for future log ins.

To set up your system so that you can log in at a graphical login screen, you must edit one file, **/etc/inittab**, by changing just one number in the runlevel section. When you are finished, reboot the computer. The next time you log in, you are presented with a graphical login prompt.

Open a shell prompt. If you are in your user account, become root by typing the **su** command.

Now, type **gedit /etc/inittab** to edit the file with **gedit**. The file **/etc/inittab** opens. Within the first screen, a section of the file which looks like the following appears:

```
# Default runlevel. The runlevels used by RHS are:
#   0 - halt (Do NOT set initdefault to this)
#   1 - Single user mode
#   2 - Multiuser, without NFS (The same as 3, if you do not have networking)
#   3 - Full multiuser mode
#   4 - unused
#   5 - X11
#   6 - reboot (Do NOT set initdefault to this)
#  id:3:initdefault:
```

To change from a console to a graphical login, you should change the number in the line **id:3:initdefault:** from a **3** to a **5**.

Warning

Change *only* the number of the default runlevel from **3** to **5**.

Your changed line should look like the following:

```
id:5:initdefault:
```

When you are satisfied with your change, save and exit the file using the **Ctrl**+**Q** keys. A window appears and asks if you would like to save the changes. Click **Save**.

The next time you log in after rebooting your system, you are presented with a graphical login prompt.

6.4.3. Problems with the X Window System (GUI)

If you are having trouble getting X (the X Window System) to start, you may not have installed it during your installation.

If you want X, you can either install the packages from the Red Hat Enterprise Linux CD-ROMs or perform an upgrade.

If you elect to upgrade, select the X Window System packages, and choose GNOME, KDE, or both, during the upgrade package selection process.

6.4.4. Problems with the X Server Crashing and Non-Root Users

If you are having trouble with the X server crashing when anyone other than root logs in, you may have a full file system (or, a lack of available hard drive space).

To verify that this is the problem you are experiencing, run the following command:

```
df -h
```

The **df** command should help you diagnose which partition is full. For additional information about **df** and an explanation of the options available (such as the **-h** option used in this example), refer to the **df** man page by typing **man df** at a shell prompt.

A key indicator is 100% full or a percentage above 90% or 95% on a partition. The **/home/** and **/tmp/** partitions can sometimes fill up quickly with user files. You can make some room on that partition by removing old files. After you free up some disk space, try running X as the user that was unsuccessful before.

6.4.5. Problems When You Try to Log In

If you did not create a user account in the **Setup Agent**, log in as root and use the password you assigned to root.

If you cannot remember your root password, boot your system as **linux single**.

Itanium users must enter boot commands with **elilo** followed by the boot command.

If you are using an x86-based system and GRUB is your installed boot loader, type **e** for edit when the GRUB boot screen has loaded. You are presented with a list of items in the configuration file for the boot label you have selected.

Choose the line that starts with **kernel** and type **e** to edit this boot entry.

At the end of the **kernel** line, add:

```
single
```

Press **Enter** to exit edit mode.

Once the boot loader screen has returned, type **b** to boot the system.

Once you have booted into single user mode and have access to the **#** prompt, you must type **passwd root**, which allows you to enter a new password for root. At this point you can type **shutdown -r now** to reboot the system with the new root password.

If you cannot remember your user account password, you must become root. To become root, type **su -** and enter your root password when prompted. Then, type **passwd <username>**. This allows you to enter a new password for the specified user account.

If the graphical login screen does not appear, check your hardware for compatibility issues. The *Hardware Compatibility List* can be found at:

```
http://hardware.redhat.com/hcl/
```

6.4.6. Is Your RAM Not Being Recognized?

Sometimes, the kernel does not recognize all of your memory (RAM). You can check this with the **cat /proc/meminfo** command.

Verify that the displayed quantity is the same as the known amount of RAM in your system. If they are not equal, add the following line to the **/boot/grub/grub.conf**:

```
mem=xxM
```

Replace *xx* with the amount of RAM you have in megabytes.

In **/boot/grub/grub.conf**, the above example would look similar to the following:

```
# NOTICE: You have a /boot partition. This means that
#  all kernel paths are relative to /boot/
default=0
timeout=30
splashimage=(hd0,0)/grub/splash.xpm.gz
title Red Hat Enterprise Linux (2.6.9-5.EL)
root (hd0,0)
kernel /vmlinuz-2.6.9-5.EL ro root=/dev/hda3 mem=128M
```

Once you reboot, the changes made to **grub.conf** are reflected on your system.

Once you have loaded the GRUB boot screen, type **e** for edit. You are presented with a list of items in the configuration file for the boot label you have selected.

Choose the line that starts with **kernel** and type **e** to edit this boot entry.

At the end of the **kernel** line, add

```
mem=xxM
```

where *xx* equals the amount of RAM in your system.

Press **Enter** to exit edit mode.

Once the boot loader screen has returned, type **b** to boot the system.

Itanium users must enter boot commands with **elilo** followed by the boot command.

Remember to replace *xx* with the amount of RAM in your system. Press **Enter** to boot.

6.4.7. Your Printer Does Not Work

If you are not sure how to set up your printer or are having trouble getting it to work properly, try using the **Printer Configuration Tool**.

Type the **system-config-printer** command at a shell prompt to launch the **Printer Configuration Tool**. If you are not root, it prompts you for the root password to continue.

6.4.8. Problems with Sound Configuration

If, for some reason, you do not hear sound and know that you do have a sound card installed, you can run the **Sound Card Configuration Tool** (**system-config-soundcard**) utility.

To use the **Sound Card Configuration Tool**, choose **Main Menu** => **System** => **Administration** => **Soundcard Detection** in GNOME, or **Main Menu** => **Administration** => **Soundcard Detection** in KDE. A small text box pops up prompting you for your root password.

You can also type the **system-config-soundcard** command at a shell prompt to launch the **Sound Card Configuration Tool**. If you are not root, it prompts you for the root password to continue.

If the **Sound Card Configuration Tool** does not work (if the sample does not play and you still do not have audio sounds), it is likely that your sound card is not yet supported in Red Hat Enterprise Linux.

6.4.9. Apache-based httpd service/Sendmail Hangs During Startup

If you are having trouble with the Apache-based **httpd** service or Sendmail hanging at startup, make sure the following line is in the **/etc/hosts** file:

```
127.0.0.1  localhost.localdomain  localhost
```

Chapter 7. Updating drivers during installation on Intel and AMD systems

In most cases, Red Hat Enterprise Linux already includes drivers for the devices that make up your system. However, if your system contains hardware that has been released very recently, drivers for this hardware might not yet be included. Sometimes, a driver update that provides support for a new device might be available from Red Hat or your hardware vendor as a *ISO image file* or a *rpm package*. Both these formats supply all the files that make up the driver update together in a single file.

Often, you do not need the new hardware during the installation process. For example, if you use a DVD to install to a local hard drive, the installation will succeed even if drivers for your network card are not available. In situations like this, complete the installation and add support for the piece of hardware afterward — refer to Section 23.1, “Driver update rpm packages” for details of using a driver update rpm package to add this support.

In other situations, you might want to add drivers for a device during the installation process to support a particular configuration. For example, you might want to install drivers for a network device or a storage adapter card to give the installer access to the storage devices that your system uses. You can use a driver update image file to add this support during installation in one of three ways:

1. place the image file in a location accessible to the installer:
   a. on a local IDE hard drive
   b. a USB storage device such as a USB flash drive
   c. on a FTP, HTTP, or NFS server on your local network (or take note of a location on the Internet where someone else has placed the image file)
2. create a *driver update disk* by unpacking the image file onto:
   a. a CD (if your computer has an IDE optical drive)
   b. a DVD (if your computer has an IDE optical drive)
   c. a floppy disk
   d. a USB storage device such as a USB flash drive
3. create an *initial ramdisk update* from the image file and store it on a PXE server. This is an advanced procedure that you should consider only if you cannot perform a driver update with any other method.

If Red Hat, your hardware vendor, or a trusted third party told you that you will require a driver update during the installation process, choose a method to supply the update from the methods described in this chapter and test it before beginning the installation. Conversely, do not perform a driver update during installation unless you are certain that your system requires it. Although installing an unnecessary driver update will not cause harm, the presence of a driver on a system for which it was not intended can complicate support.

7.1. Limitations of driver updates during installation

Unfortunately, some situations persist in which you cannot use a driver update to provide drivers during installation:

Devices already in use

You cannot use a driver update to replace drivers that the installation program has already loaded. Instead, you must complete the installation with the drivers that the installation program loaded and update to the new drivers after installation, or, if you need the new drivers for the installation process, consider performing an initial RAM disk driver update — refer to Section 7.2.3, “Preparing an initial RAM disk update”.

Devices with an equivalent device available

Because all devices of the same type are initialized together, you cannot update drivers for a device if the installation program has loaded drivers for a similar device. For example, consider a system that has two different network adapters, one of which has a driver update available. The installation program will initialize both adapters at the same time, and therefore, you will not be able to use this driver update. Again, complete the installation with the drivers loaded by the installation program and update to the new drivers after installation, or use an initial RAM disk driver update.

7.2. Preparing for a driver update during installation

If a driver update is necessary and available for your hardware, Red Hat or a trusted third party such as the hardware vendor will provide it in the form of an image file in ISO format. Some methods of performing a driver update require you to make the image file available to the installation program, others require you to use the image file to make a driver update disk, and one requires you to prepare an initial RAM disk update:

Methods that use the image file itself

- local hard drive (IDE only)
- USB storage device (for example, USB flash drive)
- network (HTTP, FTP, NFS)

Methods that use a driver update disk produced from an image file

- floppy disk
- CD (IDE only)
- DVD (IDE only)
- USB storage device (for example, USB flash drive)

Methods that use an initial RAM disk update

- PXE

Choose a method to provide the driver update, and refer to Section 7.2.1, “Preparing to use a driver update image file”, Section 7.2.2, “Preparing a driver update disk” or Section 7.2.3, “Preparing an initial RAM disk update”. Note that you can use a USB storage device either to provide an image file, or as a driver update disk.

SATA and SCSI

During installation, you cannot provide driver updates on media that is read by a device with a SATA or SCSI connection. For example, if the only optical drive on your system is a DVD drive with a SATA connection, you cannot provide a driver update on CD or DVD.

7.2.1. Preparing to use a driver update image file

7.2.1.1. Preparing to use an image file on local storage

To make the ISO image file available on local storage, such as a USB flash drive, USB hard drive, or local IDE hard drive, simply copy the file onto the storage device. You can rename the file if you find it helpful to do so, but you must not change the filename extension, which must remain **`.iso`**. In the following example, the file is named **`dd.iso`**:

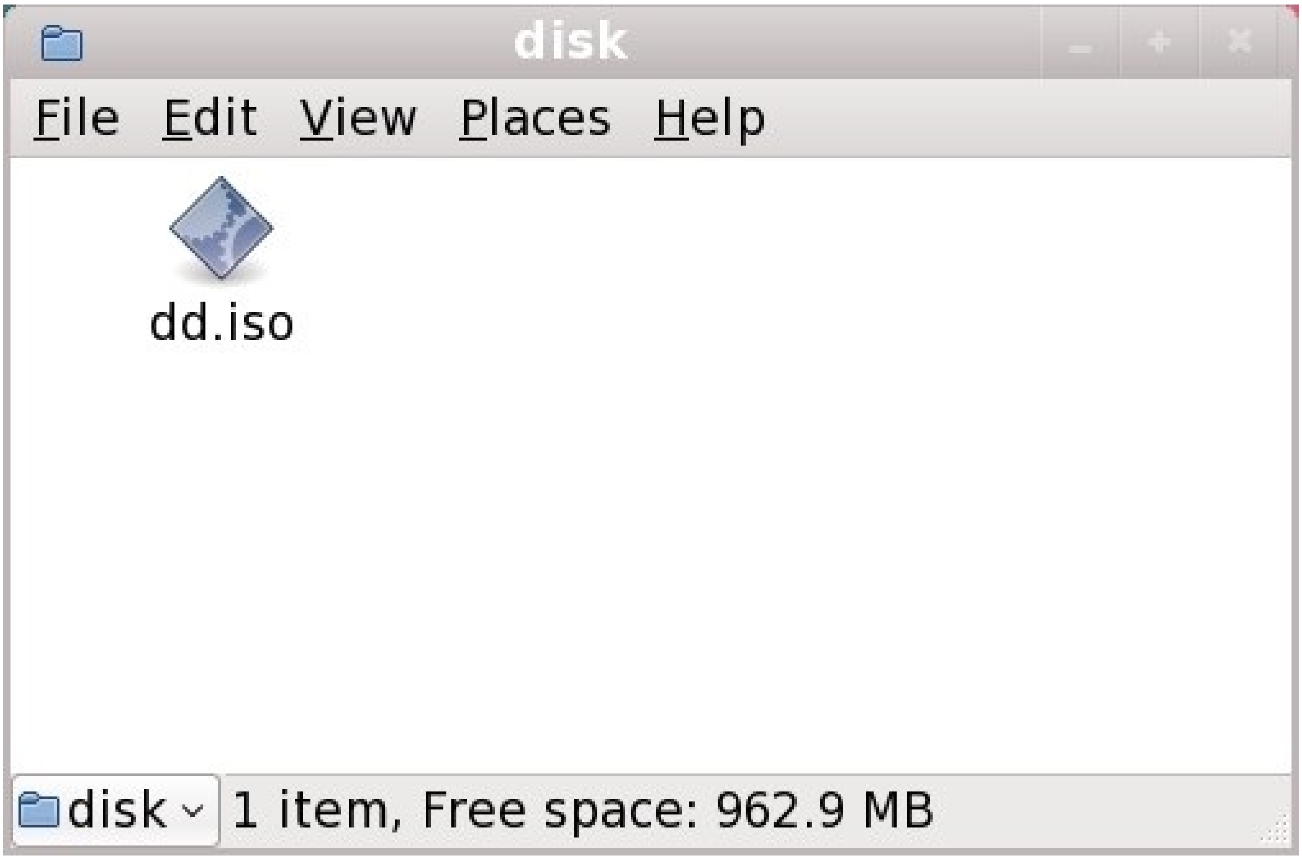

Figure 7.1. Content of a USB flash drive holding a driver update image file

Note that if you use this method, the storage device will contain only a single file. This differs from driver update disks on formats such as CD and DVD, which contain many files. The ISO image file contains all of the files that would normally be on a driver update disk.

Refer to Section 7.3.2, "Let the installer prompt you for a driver update" and Section 7.3.3, "Use a boot option to specify a driver update disk" to learn how to use the driver update disk during installation.

If you change the file system label of the device to **`OEMDRV`**, the installation program will automatically examine it for driver updates and load any that it detects. This behavior is controlled by the **`dlabel=on`** boot option, which is enabled by default. Refer to Section 7.3.1, "Let the installer automatically find a driver update disk".

7.2.1.2. Preparing to use an image file available through a network

To make the ISO image file available on a local network, place it in a publicly accessible folder on a HTTP, FTP, or NFS server. If you plan to use an image file that is already publicly available through the Internet, no special preparation is necessary. In either case, take note of the URL and verify that you can access the file from another machine on your network before commencing installation.

Refer to Section 7.3.2, “Let the installer prompt you for a driver update” and Section 7.3.4, “Use a boot option to specify a driver update image file on a network” to learn how to specify this network location during installation.

7.2.2. Preparing a driver update disk

You can use a variety of media to create a driver update disk, including CD, DVD, floppy disk, and USB storage devices such as USB flash drives

7.2.2.1. Creating a driver update disk on CD or DVD

These instructions assume that you use the GNOME desktop

CD/DVD Creator is part of the GNOME desktop. If you use a different Linux desktop, or a different operating system altogether, you will need to use another piece of software to create the CD or DVD. The steps will be generally similar.

Make sure that the software that you choose can create CDs or DVDs from image files. While this is true of most CD and DVD burning software, exceptions exist. Look for a button or menu entry labeled **`burn from image`** or similar. If your software lacks this feature, or you do not select it, the resulting disk will hold only the image file itself, instead of the contents of the image file.

1. Use the desktop file manager to locate the driver update ISO image file supplied to you by Red Hat or your hardware vendor.

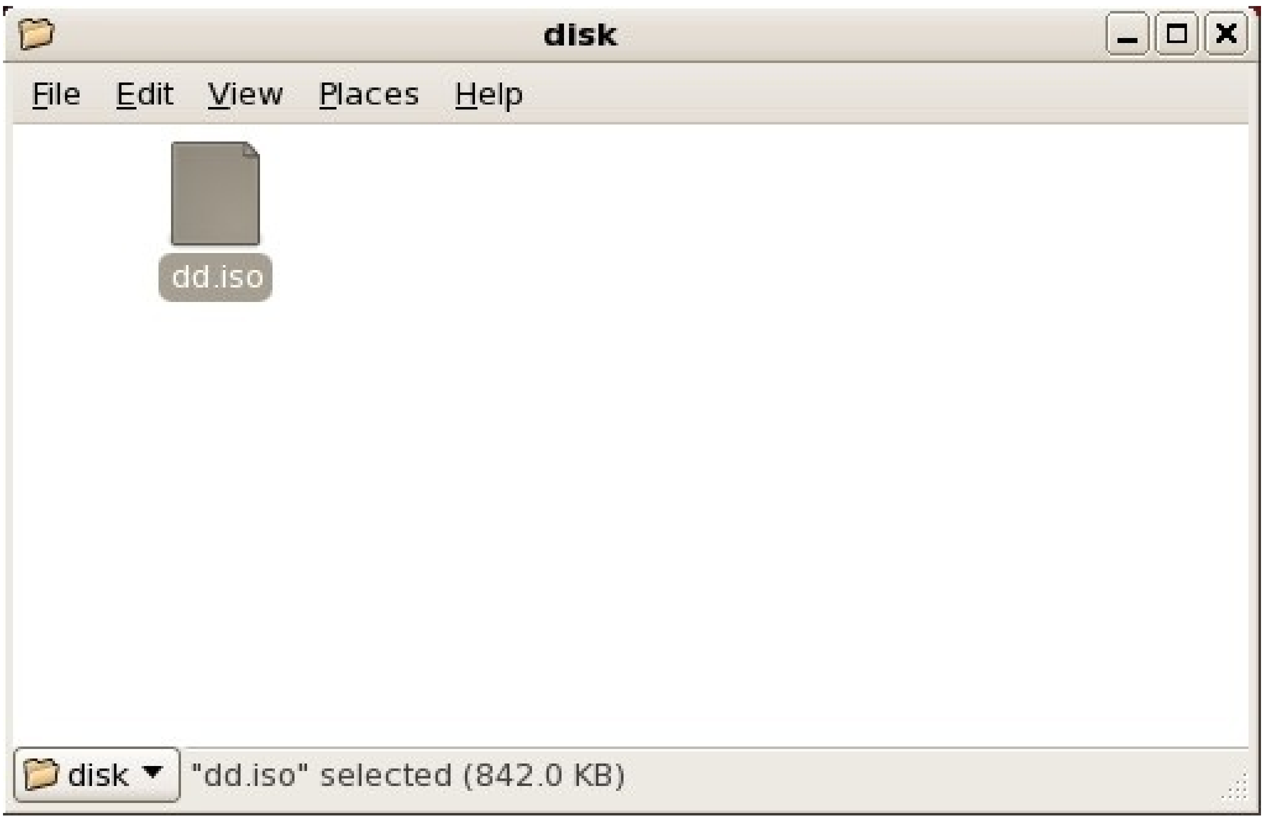

Figure 7.2. A typical .iso file displayed in a file manager window

2. Right-click on this file and choose **`Write to disc`**. You will see a window similar to the following:

Figure 7.3. CD/DVD Creator's Write to Disc dialog

3. Click the **`Write`** button. If a blank disc is not already in the drive, **CD/DVD Creator** will prompt you to insert one.

After you burn a driver update disk CD or DVD, verify that the disk was created successfully by inserting it into your system and browsing to it using the file manager. You should see a list of files similar to the following:

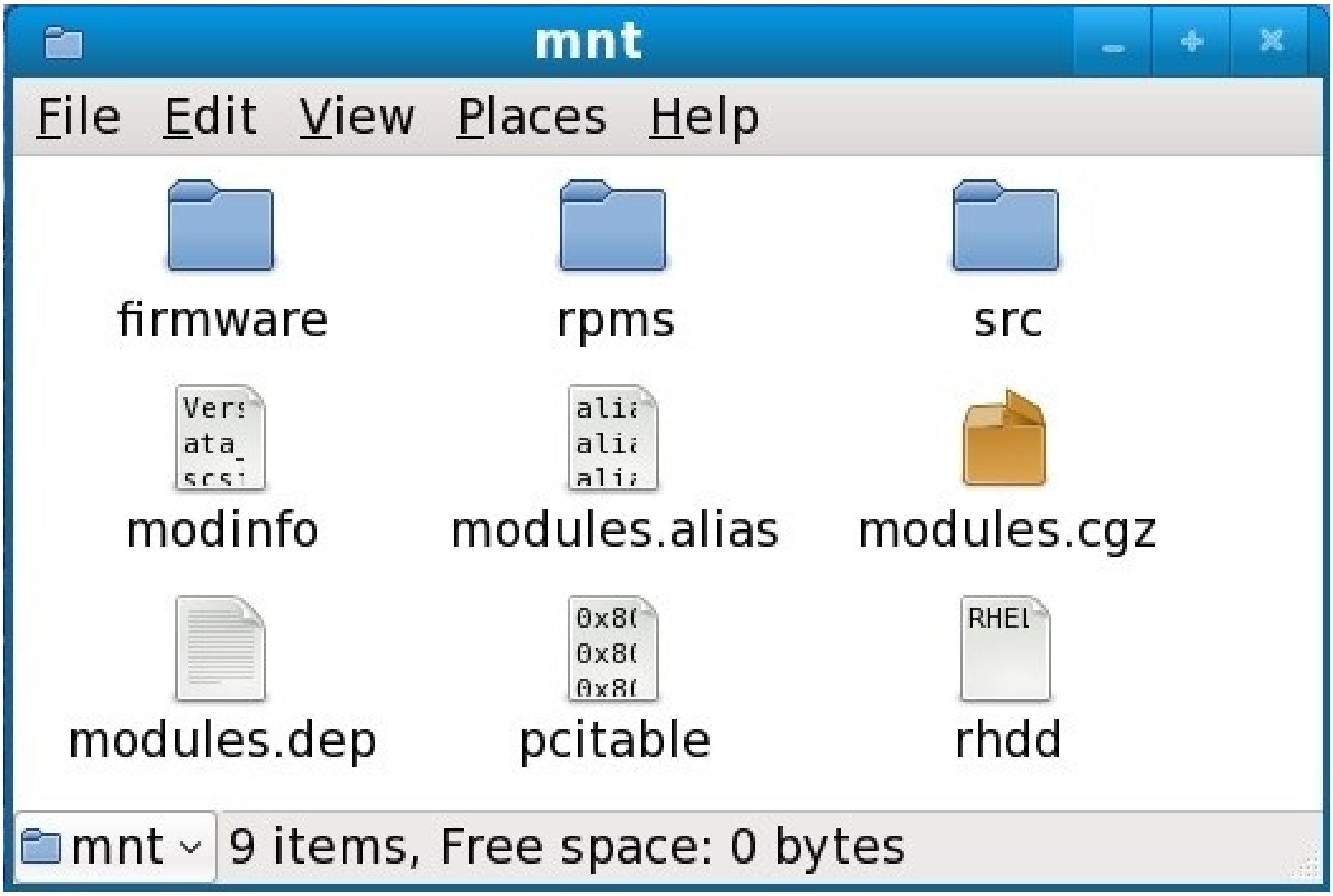

Figure 7.4. Contents of a typical driver update disc on CD or DVD

If you see only a single file ending in **.iso**, then you have not created the disk correctly and should try again. Ensure that you choose an option similar to **burn from image** if you use a Linux desktop other than GNOME or if you use a different operating system.

Refer to Section 7.3.2, “Let the installer prompt you for a driver update” and Section 7.3.3, “Use a boot option to specify a driver update disk” to learn how to use the driver update disk during installation.

7.2.2.2. Creating a driver update disk on floppy disk, or USB storage device

These instructions assume that you use Linux

The following procedure assumes that you are creating a driver update disk using Linux. To create a driver update disk using a different operating system, you will need to find a tool that can extract files from ISO images. You must then place the extracted files on a removable disk or USB storage device.

These instructions could destroy data

If you perform this procedure with a disk or USB storage device that already contains data, this data will be destroyed with no warning. Make sure that you specify the correct disk or USB storage device, and make sure that this disk or storage device does not contain any data that you want to keep.

1. Insert a blank, formatted floppy disk into an available drive, or connect an empty USB storage device (such as a USB flash drive) to your computer. Note the device name allocated to this disk, for example, **/dev/fd0** for a floppy disk in the first floppy drive on your system.

   If you do not know the device name, become root and use the command **fdisk -l** on the command line. You will see a list of all storage devices available on your system. Compare the output of **fdisk -l** when the disk inserted or the storage device is attached with the output of this command when the disk is removed or the storage device is disconnected.

2. At the command line, change into the directory that contains the image file.

3. At the command line, type:

   ```
   dd if=image of=device
   ```

   where *image* is the image file, and *device* is the device name. For example, to create a driver disk on floppy disk **/dev/fd0** from driver update image file **dd.iso**, you would use:

   ```
   dd if=dd.iso of=/dev/fd0
   ```

After you create a driver update disk, insert it (if you used a disk) or attach it (if you used a USB storage device) and browse to it using the file manager. If you see a list of files similar to those illustrated in Figure 7.4, “Contents of a typical driver update disc on CD or DVD”, you know that you have created the driver update disk correctly.

Refer to Section 7.3.2, “Let the installer prompt you for a driver update” and Section 7.3.3, “Use a boot option to specify a driver update disk” to learn how to use the driver update disk during installation.

7.2.3. Preparing an initial RAM disk update

Advanced procedure

This is an advanced procedure that you should consider only if you cannot perform a driver update with any other method.

The Red Hat Enterprise Linux installation program can load updates for itself early in the installation process from a *RAM disk* — an area of your computer's memory that temporarily behaves as if it were a disk. You can use this same capability to load driver updates. To perform a driver update during installation, your computer must be able to boot from a *preboot execution environment* (PXE) server, and you must have a PXE server available on your network. Refer to Chapter 34, *PXE Network Installations* for instructions on using PXE during installation.

To make the driver update available on your PXE server:

1. Place the driver update image file on your PXE server. Usually, you would do this by downloading it to the PXE server from a location on the Internet specified by Red Hat or your hardware vendor. Names of driver update image files end in **.iso**.

2. Copy the driver update image file into the **/tmp/initrd_update** directory.

3. Rename the driver update image file to **dd.img**.

4. At the command line, change into the **/tmp/initrd_update** directory, type the following command, and press **Enter**:

```
find . | cpio --quiet -c -o | gzip -9 >/tmp/initrd_update.img
```

5. Copy the file **/tmp/initrd_update.img** into the directory the holds the target that you want to use for installation. This directory is placed under the **/tftpboot/pxelinux/** directory. For example, **/tftpboot/pxelinux/r5su3/** might hold the PXE target for Red Hat Enterprise Linux 5.3 Server.

6. Edit the **/tftpboot/pxelinux/pxelinux.cfg/default** file to include an entry that includes the initial RAM disk update that you just created, in the following format:

```
label target-dd
kernel target/vmlinuz
append initrd=target/initrd.img,target/dd.img
```

Where *target* is the target that you want to use for installation.

Example 7.1. Preparing an initial RAM disk update from a driver update image file

In this example, **driver_update.iso** is a driver update image file that you downloaded from the Internet to a directory on your PXE server. The target that you want to PXE boot from is located in **/tftpboot/pxelinux/r5su3**

At the command line, change to the directory that holds the file and enter the following commands:

```
$ cp driver_update.iso /tmp/initrd_update/dd.img
$ cd /tmp/initrd_update
$ find . | cpio --quiet -c -o | gzip -9 >/tmp/initrd_update.img
$ cp /tmp/initrd_update.img /tftpboot/pxelinux/r5su3/dd.img
```

Edit the **/tftpboot/pxelinux/pxelinux.cfg/default** file and include the following entry:

```
label r5su3-dd
kernel r5su3/vmlinuz
append initrd=r5su3/initrd.img,r5su3/dd.img
```

Refer to Section 7.3.5, “Select a PXE target that includes a driver update” to learn how to use an initial RAM disk update during installation.

7.3. Performing a driver update during installation

You can perform a driver update during installation in the following ways:

- let the installer automatically find a driver update disk.
- let the installer prompt you for a driver update.
- use a boot option to specify a driver update disk.
- use a boot option to specify a driver update image file on a network.
- select a PXE target that includes a driver update.

7.3.1. Let the installer automatically find a driver update disk

Attach a block device with the filesystem label **OEMDRV** before starting the installation process. The

installer will automatically examine the device and load any driver updates that it detects and will not prompt you during the process. Refer to Section 7.2.1.1, “Preparing to use an image file on local storage” to prepare a storage device for the installer to find.

7.3.2. Let the installer prompt you for a driver update

1. Begin the installation normally for whatever method you have chosen. If the installer cannot load drivers for a piece of hardware that is essential for the installation process (for example, if it cannot detect any network or storage controllers), it prompts you to insert a driver update disk:

Figure 7.5. The no driver found dialog

2. Select **`Use a driver disk`** and refer to Section 7.4, “Specifying the location of a driver update image file or driver update disk”.

7.3.3. Use a boot option to specify a driver update disk

> **Choose this method only for completely new drivers**
>
> This method only works to introduce completely new drivers, not to update existing drivers.

1. Type **`linux dd`** at the boot prompt at the start of the installation process and press **`Enter`**. The installer prompts you to confirm that you have a driver disk:

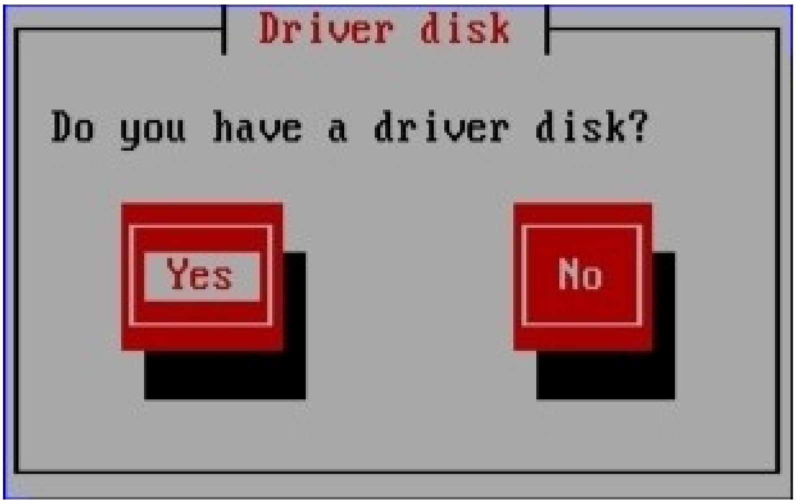

Figure 7.6. The driver disk prompt

2. Insert the driver update disk that you created on CD, DVD, floppy disk, or USB storage device and select **Yes**. The installer examines the storage devices that it can detect. If there is only one possible location that could hold a driver disk (for example, the installer detects the presence of a floppy disk, but no other storage devices) it will automatically load any driver updates that it finds at this location.

   If the installer finds more than one location that could hold a driver update, it prompts you to specify the location of the update. Refer to to Section 7.4, “Specifying the location of a driver update image file or driver update disk” .

7.3.4. Use a boot option to specify a driver update image file on a network

Choose this method only for completely new drivers

This method only works to introduce completely new drivers, not to update existing drivers.

Type **`linux dd=`*`URL`*** (where *URL* is the HTTP, FTP, or NFS address of a driver update image) at the boot prompt at the start of the installation process and press **Enter**. The installer will retrieve the driver update image from that address and use it during installation.

7.3.5. Select a PXE target that includes a driver update

1. Select **`network boot`** in your computer's BIOS or boot menu. The procedure to specify this option varies widely among different computers. Consult your hardware documentation or the hardware vendor for specifics relevant to your computer.

2. In the preexecution boot environment (PXE), choose the boot target that you prepared on your PXE server. For example, if you labeled this environment **`r5su3-dd`** in the **`/tftpboot/pxelinux/pxelinux.cfg/default`** file on your PXE server, type **`r5su3-dd`** at the prompt and press **Enter**.

Refer to Section 7.2.3, “Preparing an initial RAM disk update” and Chapter 34, *PXE Network Installations* for instructions on using PXE to perform an update during installation. Note that this is an advanced procedure — do not attempt it unless other methods of performing a driver update fail.

7.4. Specifying the location of a driver update image file or driver update disk

If the installer detects more than one possible device that could hold a driver update, it prompts you to select the correct device. If you are not sure which option represents the device on which the driver update is stored, try the various options in order until you find the correct one.

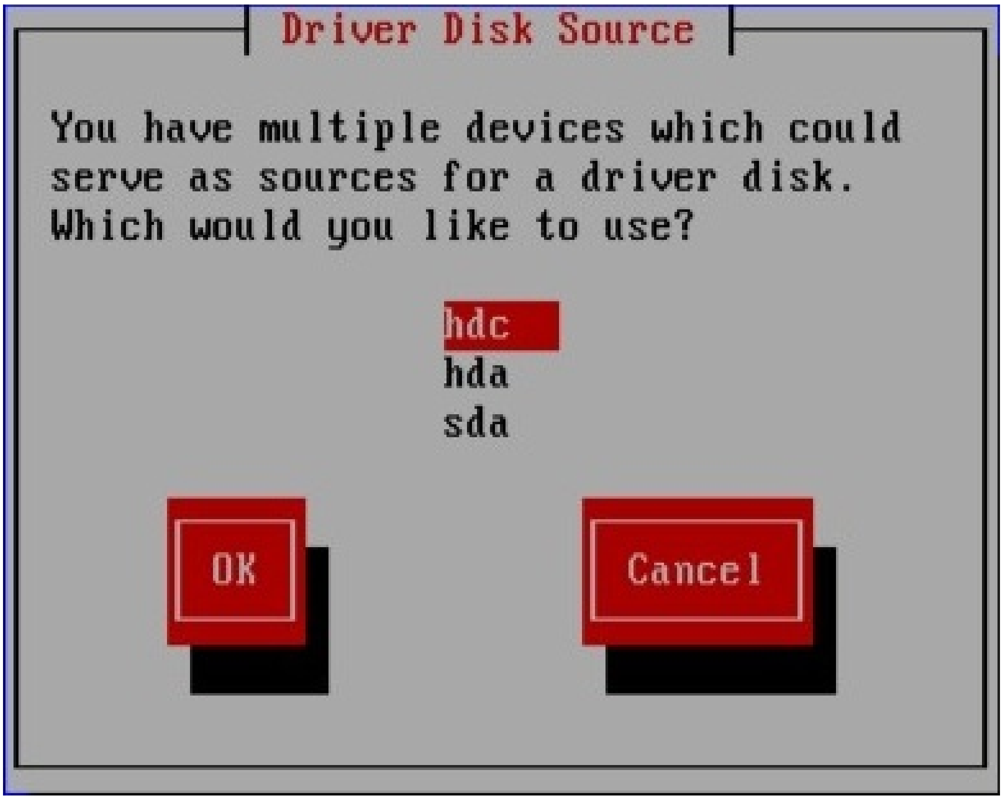

Figure 7.7. Selecting a driver disk source

If the device that you choose contains no suitable update media, the installer will prompt you to make another choice.

If you made a driver update disk on floppy disk, CD, DVD, or USB storage device, the installer now loads the driver update. However, if the device that you selected is a type of device that could contain more than one partition (whether the device currently has more than one partition or not), the installer might prompt you to select the partition that holds the driver update.

Figure 7.8. Selecting a driver disk partition

The installer prompts you to specify which file contains the driver update:

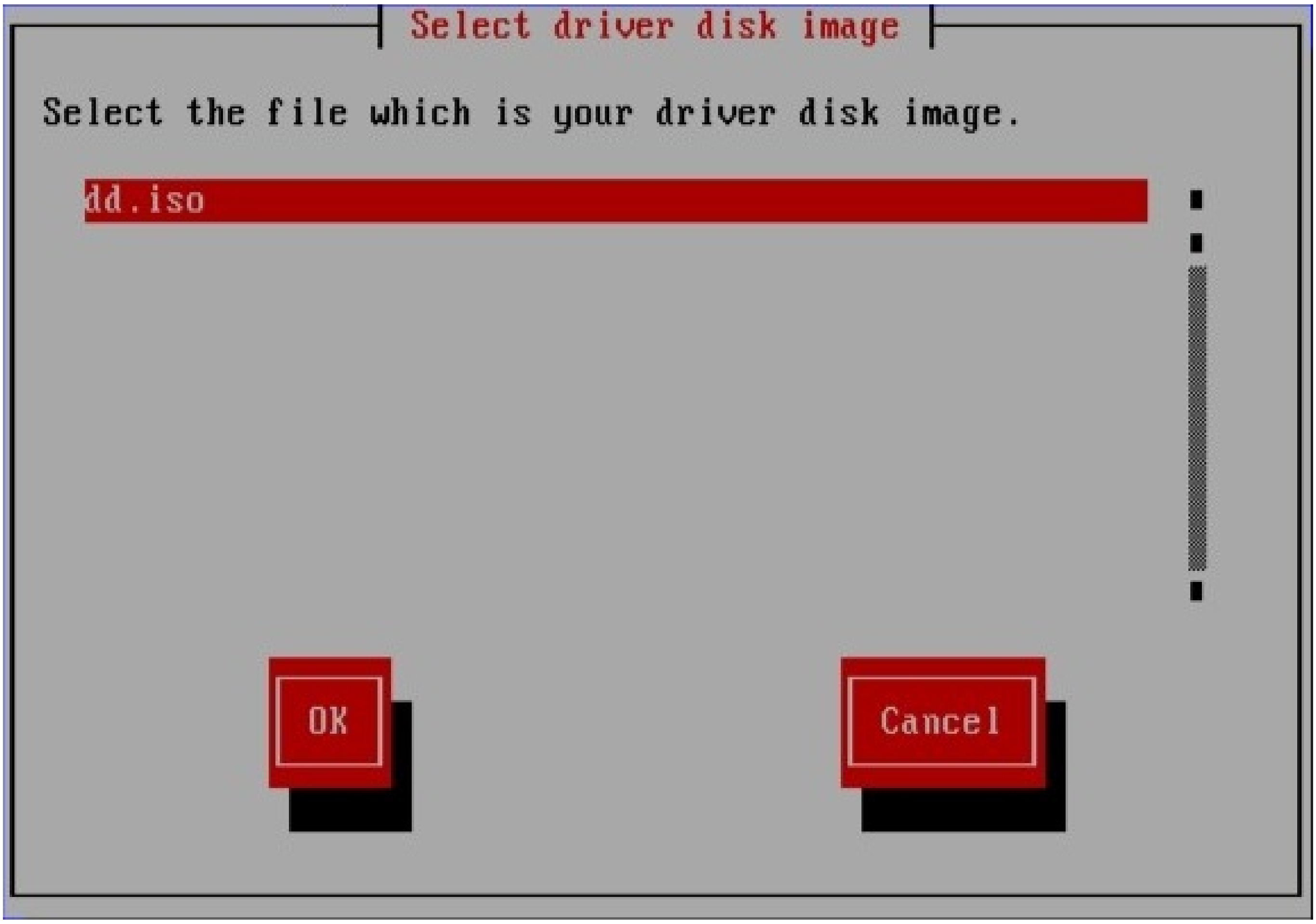

Figure 7.9. Selecting an ISO image

Expect to see these screens if you stored the driver update on an internal hard drive or on a USB storage device. You should not see them if the driver update is on a floppy disk, CD, or DVD.

Regardless of whether you are providing a driver update in the form of an image file or with a driver update disk, the installer now copies the appropriate update files into a temporary storage area (located in system RAM and not on disk). The installer might ask whether you would like to use additional driver updates. If you select **Yes**, you can load additional updates in turn. When you have no further driver updates to load, select **No**. If you stored the driver update on removeable media, you can now safely eject or disconnect the disk or device. The installer no longer requires the driver update, and you can re-use the media for other purposes.

Chapter 8. Additional Boot Options for Intel® and AMD Systems

This section discusses additional boot and kernel boot options available for the Red Hat Enterprise Linux installation program.

To use any of the boot options presented here, type the command you wish to invoke at the installation **`boot:`** prompt.

Boot Time Command Arguments

`askmethod`

this command asks you to select the installation method you would like to use when booting from the Red Hat Enterprise Linux CD-ROM.

`apic`

this x86 boot command works around a bug commonly encountered in the Intel 440GX chipset BIOS and should only be executed with the installation program kernel.

`dd`

this argument causes the installation program to prompt you to use a driver diskette.

`dd=`*`url`*

this argument causes the installation program to prompt you to use a driver image from a specified HTTP, FTP, or NFS network address.

`display=`*`ip`*`:0`

this command allows remote display forwarding. In this command, *ip* should be replaced with the IP address of the system on which you want the display to appear.

On the system you want the display to appear on, you must execute the command **`xhost +`*`remotehostname`***, where *remotehostname* is the name of the host from which you are running the original display. Using the command **`xhost +`*`remotehostname`*** limits access to the remote display terminal and does not allow access from anyone or any system not specifically authorized for remote access.

`driverdisk`

this command performs the same function as the **`dd`** command and also prompts you to use a driver diskette during the installation of Red Hat Enterprise Linux.

`linux upgradeany`

this command relaxes some of the checks on your **`/etc/redhat-release`** file. If your **`/etc/redhat-release`** file has been changed from the default, your Red Hat Enterprise Linux installation may not be found when attempting an upgrade to Red Hat Enterprise Linux 5. Use this option only if your existing Red Hat Enterprise Linux installation was not detected.

`mediacheck`

this command gives you the option of testing the integrity of the install source (if an ISO-based method). this command works with the CD, DVD, hard drive ISO, and NFS ISO installation methods. Verifying that the ISO images are intact before you attempt an installation helps to avoid problems that are often encountered during an installation.

`mem=`*`xxx`*`m`

this command allows you to override the amount of memory the kernel detects for the machine. This may be needed for some older systems where only 16 mb is detected and for some new machines where the video card shares the video memory with the main memory. When executing this command, *xxx* should be replaced with the amount of memory in megabytes.

`mpath`

enables multipath support.

> **Important — mandatory for installation on multipath devices**
>
> If you install Red Hat Enterprise Linux 5.11 on a network storage device accessible through multiple paths, you must boot the installation process with this option. If you do not specify this option at boot time, installation will fail, or the system will fail to boot after installation completes.

`nmi_watchdog=1`

this command enables the built-in kernel deadlock detector. This command can be used to debug hard kernel lockups. by executing periodic NMI (Non Maskable Interrupt) interrupts, the kernel can monitor whether any CPU has locked up and print out debugging messages as needed.

`noapic`

this x86 boot command tells the kernel not to use the APIC chip. It may be helpful for some motherboards with a bad APIC (such as the Abit BP6) or with a buggy bios. systems based on the nvidia nforce3 chipset (such as the Asus SK8N) have been known to hang during IDE detection at boot time, or display other interrupt-delivery issues.

`noeject`

do not eject optical discs after installation. This option is useful in remote installations where it is difficult to close the tray afterwards.

`nomce`

this x86 boot command disables self-diagnosis checks performed on the CPU. the kernel enables self-diagnosis on the CPU by default (called *machine check exception*). Early Compaq Pentium systems may need this option as they do not support processor error checking correctly. A few other laptops, notably those using the Radeon IGP chipset, may also need this option.

`nonet`

this command disables network hardware probing.

nopass

this command disables the passing of keyboard and mouse information to stage 2 of the installation program. It can be used to test keyboard and mouse configuration screens during stage 2 of the installation program when performing a network installation.

nopcmcia

this command ignores any PCMCIA controllers in system.

noprobe

this command disables hardware detection and instead prompts the user for hardware information.

noshell

this command disables shell access on virtual console 2 during an installation.

nostorage

this command disables probing for SCSI and RAID storage hardware.

nousb

this command disables the loading of USB support during the installation. If the installation program tends to hang early in the process, this command may be helpful.

nousbstorage

this command disables the loading of the usbstorage module in the installation program's loader. It may help with device ordering on SCSI systems.

numa=off

Red Hat Enterprise Linux supports NUMA (non-uniform memory access) on the AMD64 architecture. while all cpus can access all memory even without numa support, the numa support present in the updated kernel causes memory allocations to favor the cpu on which they originate as much as possible, thereby minimizing inter-CPU memory traffic. This can provide significant performance improvements in certain applications. to revert to the original non-NUMA behavior, specify this boot option.

reboot=b

this x86, AMD64, and Intel® EM64T boot command changes the way the kernel tries to reboot the machine. If a kernel hang is experienced while the system is shutting down, this command may cause the system to reboot successfully.

rescue

this command runs rescue mode. Refer to Chapter 27, *Basic System Recovery* for more information about rescue mode.

resolution=

tells the installation program which video mode to run. it accepts any standard resolution, such as **640x480**, **800x600**, **1024x768**, and so on.

serial

this command turns on serial console support.

`text`

this command disables the graphical installation program and forces the installation program to run in text mode.

`updates`

this command prompts you to insert a floppy diskette containing updates (bug fixes) for the anaconda installation program. It is not needed if you are performing a network installation and have already placed the updates image contents in **`rhupdates/`** on the server.

`updates=`

this command allows you to specify a URL to retrieve updates (bug fixes) for the anaconda installation program.

`vnc`

this command allows you to install from a VNC server.

`vncpassword=`

this command sets the password used to connect to the VNC server.

Important

Other kernel boot options have no particular meaning for **anaconda** and do not affect the installation process. However, if you use these options to boot the installation system, **anaconda** will preserve them in the bootloader configuration.

Chapter 9. The GRUB Boot Loader

When a computer with Red Hat Enterprise Linux is turned on, the operating system is loaded into memory by a special program called a *boot loader*. A boot loader usually exists on the system's primary hard drive (or other media device) and has the sole responsibility of loading the Linux kernel with its required files or (in some cases) other operating systems into memory.

9.1. Boot Loaders and System Architecture

Each architecture capable of running Red Hat Enterprise Linux uses a different boot loader. The following table lists the boot loaders available for each architecture:

Table 9.1. Boot Loaders by Architecture

Architecture	Boot Loaders
AMD® AMD64	GRUB
IBM® eServer™ System i™	OS/400®
IBM® eServer™ System p™	YABOOT
IBM® System z®	z/IPL
IBM® System z®	z/IPL
Intel® Itanium™	ELILO
x86	GRUB

This chapter discusses commands and configuration options for the GRUB boot loader included with Red Hat Enterprise Linux for the x86 architecture.

9.2. GRUB

The *GNU GRand Unified Boot loader* (GRUB) is a program which enables the selection of the installed operating system or kernel to be loaded at system boot time. It also allows the user to pass arguments to the kernel.

9.2.1. GRUB and the x86 Boot Process

This section discusses the specific role GRUB plays when booting an x86 system. For a look at the overall boot process, refer to Section 33.2, “A Detailed Look at the Boot Process”.

GRUB loads itself into memory in the following stages:

1. *The Stage 1 or primary boot loader is read into memory by the BIOS from the MBR* [4]. The primary boot loader exists on less than 512 bytes of disk space within the MBR and is capable of loading either the Stage 1.5 or Stage 2 boot loader.

2. *The Stage 1.5 boot loader is read into memory by the Stage 1 boot loader, if necessary.* Some hardware requires an intermediate step to get to the Stage 2 boot loader. This is sometimes true when the **`/boot/`** partition is above the 1024 cylinder head of the hard drive or when using LBA mode. The Stage 1.5 boot loader is found either on the **`/boot/`** partition or on a small part of the MBR and the **`/boot/`** partition.

3. *The Stage 2 or secondary boot loader is read into memory.* The secondary boot loader displays the GRUB menu and command environment. This interface allows the user to select which kernel or operating system to boot, pass arguments to the kernel, or look at system parameters.

4. *The secondary boot loader reads the operating system or kernel as well as the contents of* ***/boot/sysroot/*** *into memory.* Once GRUB determines which operating system or kernel to start, it loads it into memory and transfers control of the machine to that operating system.

The method used to boot Red Hat Enterprise Linux is called *direct loading* because the boot loader loads the operating system directly. There is no intermediary between the boot loader and the kernel.

The boot process used by other operating systems may differ. For example, the Microsoft® Windows® operating system, as well as other operating systems, are loaded using *chain loading*. Under this method, the MBR points to the first sector of the partition holding the operating system, where it finds the files necessary to actually boot that operating system.

GRUB supports both direct and chain loading boot methods, allowing it to boot almost any operating system.

Warning

During installation, Microsoft's DOS and Windows installation programs completely overwrite the MBR, destroying any existing boot loaders. If creating a dual-boot system, it is best to install the Microsoft operating system first.

9.2.2. Features of GRUB

GRUB contains several features that make it preferable to other boot loaders available for the x86 architecture. Below is a partial list of some of the more important features:

- *GRUB provides a true command-based, pre-OS environment on x86 machines.* This feature affords the user maximum flexibility in loading operating systems with specified options or gathering information about the system. For years, many non-x86 architectures have employed pre-OS environments that allow system booting from a command line.

- *GRUB supports Logical Block Addressing (LBA) mode.* LBA places the addressing conversion used to find files in the hard drive's firmware, and is used on many IDE and all SCSI hard devices. Before LBA, boot loaders could encounter the 1024-cylinder BIOS limitation, where the BIOS could not find a file after the 1024 cylinder head of the disk. LBA support allows GRUB to boot operating systems from partitions beyond the 1024-cylinder limit, so long as the system BIOS supports LBA mode. Most modern BIOS revisions support LBA mode.

- *GRUB can read ext2 partitions.* This functionality allows GRUB to access its configuration file, **/boot/grub/grub.conf**, every time the system boots, eliminating the need for the user to write a new version of the first stage boot loader to the MBR when configuration changes are made. The only time a user needs to reinstall GRUB on the MBR is if the physical location of the **/boot/** partition is moved on the disk. For details on installing GRUB to the MBR, refer to Section 9.3, “Installing GRUB”.

9.3. Installing GRUB

If GRUB was not installed during the installation process, it can be installed afterward. Once installed, it automatically becomes the default boot loader.

Before installing GRUB, make sure to use the latest GRUB package available or use the GRUB package from the installation CD-ROMs. For instructions on installing packages, see the chapter titled *Package Management with RPM* in the Red Hat Enterprise Linux Deployment Guide.

Once the GRUB package is installed, open a root shell prompt and run the command **`/sbin/grub-install <location>`**, where *<location>* is the location that the GRUB Stage 1 boot loader should be installed. For example, the following command installs GRUB to the MBR of the master IDE device on the primary IDE bus:

```
/sbin/grub-install /dev/hda
```

The next time the system boots, the GRUB graphical boot loader menu appears before the kernel loads into memory.

Important — GRUB and RAID

GRUB cannot construct a software RAID. Therefore, the **`/boot`** directory must reside on a single, specific disk partition. The **`/boot`** directory cannot be striped across multiple disks, as in a level 0 RAID. To use a level 0 RAID on your system, place /boot on a separate partition outside the RAID.

Similarly, because the **`/boot`** directory must reside on a single, specific disk partition, **GRUB** cannot boot the system if the disk holding that partition fails or is removed from the system. This is true even if the disk is mirrored in a level 1 RAID. The following Red Hat Knowledgebase article describes how to make the system bootable from another disk in the mirrored set: http://kbase.redhat.com/faq/docs/DOC-7095

Note that these issues apply only to RAID that is implemented in software, where the individual disks that make up the array are still visible as individual disks on the system. These issues do not apply to hardware RAID where multiple disks are represented as a single device.

9.4. GRUB Terminology

One of the most important things to understand before using GRUB is how the program refers to devices, such as hard drives and partitions. This information is particularly important when configuring GRUB to boot multiple operating systems.

9.4.1. Device Names

When referring to a specific device with GRUB, do so using the following format (note that the parentheses and comma are very important syntactically):

(*<type-of-device><bios-device-number>*,*<partition-number>*)

The *<type-of-device>* specifies the type of device from which GRUB boots. The two most common options are **`hd`** for a hard disk or **`fd`** for a 3.5 diskette. A lesser used device type is also available called **`nd`** for a network disk. Instructions on configuring GRUB to boot over the network are available online at http://www.gnu.org/software/grub/manual/.

The *<bios-device-number>* is the BIOS device number. The primary IDE hard drive is numbered **0** and a secondary IDE hard drive is numbered **1**. This syntax is roughly equivalent to that used for devices by the kernel. For example, the **a** in **hda** for the kernel is analogous to the **0** in **hd0** for GRUB, the **b** in **hdb** is analogous to the **1** in **hd1**, and so on.

The *<partition-number>* specifies the number of a partition on a device. Like the *<bios-device-number>*, most types of partitions are numbered starting at **0**. However, BSD partitions are specified using letters, with **a** corresponding to **0**, **b** corresponding to **1**, and so on.

Note

The numbering system for devices under GRUB always begins with **0**, not **1**. Failing to make this distinction is one of the most common mistakes made by new users.

To give an example, if a system has more than one hard drive, GRUB refers to the first hard drive as **(hd0)** and the second as **(hd1)**. Likewise, GRUB refers to the first partition on the first drive as **(hd0,0)** and the third partition on the second hard drive as **(hd1,2)**.

In general the following rules apply when naming devices and partitions under GRUB:

- It does not matter if system hard drives are IDE or SCSI, all hard drives begin with the letters **hd**. The letters **fd** are used to specify 3.5 diskettes.
- To specify an entire device without respect to partitions, leave off the comma and the partition number. This is important when telling GRUB to configure the MBR for a particular disk. For example, **(hd0)** specifies the MBR on the first device and **(hd3)** specifies the MBR on the fourth device.
- If a system has multiple drive devices, it is very important to know how the drive boot order is set in the BIOS. This is a simple task if a system has only IDE or SCSI drives, but if there is a mix of devices, it becomes critical that the type of drive with the boot partition be accessed first.

9.4.2. File Names and Blocklists

When typing commands to GRUB that reference a file, such as a menu list, it is necessary to specify an absolute file path immediately after the device and partition numbers.

The following illustrates the structure of such a command:

(*<device-type><device-number>*,*<partition-number>*)*</path/to/file>*

In this example, replace *<device-type>* with **hd**, **fd**, or **nd**. Replace *<device-number>* with the integer for the device. Replace *</path/to/file>* with an absolute path relative to the top-level of the device.

It is also possible to specify files to GRUB that do not actually appear in the file system, such as a chain loader that appears in the first few blocks of a partition. To load such files, provide a *blocklist* that specifies block by block where the file is located in the partition. Since a file is often comprised of several different sets of blocks, blocklists use a special syntax. Each block containing the file is specified by an offset number of blocks, followed by the number of blocks from that offset point. Block offsets are listed sequentially in a comma-delimited list.

The following is a sample blocklist:

```
0+50,100+25,200+1
```

This sample blocklist specifies a file that starts at the first block on the partition and uses blocks 0 through 49, 100 through 124, and 200.

Knowing how to write blocklists is useful when using GRUB to load operating systems which require chain loading. It is possible to leave off the offset number of blocks if starting at block 0. As an example, the chain loading file in the first partition of the first hard drive would have the following name:

```
(hd0,0)+1
```

The following shows the **chainloader** command with a similar blocklist designation at the GRUB command line after setting the correct device and partition as root:

```
chainloader +1
```

9.4.3. The Root File System and GRUB

The use of the term *root file system* has a different meaning in regard to GRUB. It is important to remember that GRUB's root file system has nothing to do with the Linux root file system.

The GRUB root file system is the top level of the specified device. For example, the image file **(hd0,0)/grub/splash.xpm.gz** is located within the **/grub/** directory at the top-level (or root) of the **(hd0,0)** partition (which is actually the **/boot/** partition for the system).

Next, the **kernel** command is executed with the location of the kernel file as an option. Once the Linux kernel boots, it sets up the root file system that Linux users are familiar with. The original GRUB root file system and its mounts are forgotten; they only existed to boot the kernel file.

Refer to the **root** and **kernel** commands in Section 9.6, “GRUB Commands” for more information.

9.5. GRUB Interfaces

GRUB features three interfaces which provide different levels of functionality. Each of these interfaces allows users to boot the Linux kernel or another operating system.

The interfaces are as follows:

> **Note**
>
> The following GRUB interfaces can only be accessed by pressing any key within the three seconds of the GRUB menu bypass screen.

Menu Interface

This is the default interface shown when GRUB is configured by the installation program. A menu of operating systems or preconfigured kernels are displayed as a list, ordered by name. Use the arrow keys to select an operating system or kernel version and press the **Enter** key to boot it. If you do nothing on this screen, then after the time out period expires GRUB will load the default option.

Press the **e** key to enter the entry editor interface or the **c** key to load a command line interface.

Refer to Section 9.7, “GRUB Menu Configuration File” for more information on configuring this interface.

Menu Entry Editor Interface

To access the menu entry editor, press the **e** key from the boot loader menu. The GRUB commands for that entry are displayed here, and users may alter these command lines before booting the operating system by adding a command line (**o** inserts a new line after the current line and **O** inserts a new line before it), editing one (**e**), or deleting one (**d**).

After all changes are made, the **b** key executes the commands and boots the operating system. The **Esc** key discards any changes and reloads the standard menu interface. The **c** key loads the command line interface.

Note

For information about changing runlevels using the GRUB menu entry editor, refer to Section 9.8, “Changing Runlevels at Boot Time”.

Command Line Interface

The command line interface is the most basic GRUB interface, but it is also the one that grants the most control. The command line makes it possible to type any relevant GRUB commands followed by the **Enter** key to execute them. This interface features some advanced shell-like features, including **Tab** key completion based on context, and **Ctrl** key combinations when typing commands, such as **Ctrl**+**a** to move to the beginning of a line and **Ctrl**+**e** to move to the end of a line. In addition, the arrow, **Home**, **End**, and **Delete** keys work as they do in the **bash** shell.

Refer to Section 9.6, “GRUB Commands” for a list of common commands.

9.5.1. Interfaces Load Order

When GRUB loads its second stage boot loader, it first searches for its configuration file. Once found, the menu interface bypass screen is displayed. If a key is pressed within three seconds, GRUB builds a menu list and displays the menu interface. If no key is pressed, the default kernel entry in the GRUB menu is used.

If the configuration file cannot be found, or if the configuration file is unreadable, GRUB loads the command line interface, allowing the user to type commands to complete the boot process.

If the configuration file is not valid, GRUB prints out the error and asks for input. This helps the user see precisely where the problem occurred. Pressing any key reloads the menu interface, where it is then possible to edit the menu option and correct the problem based on the error reported by GRUB. If the correction fails, GRUB reports an error and reloads the menu interface.

9.6. GRUB Commands

GRUB allows a number of useful commands in its command line interface. Some of the commands accept options after their name; these options should be separated from the command and other options on that line by space characters.

The following is a list of useful commands:

- **`boot`** — Boots the operating system or chain loader that was last loaded.
- **`chainloader` *`</path/to/file>`*** — Loads the specified file as a chain loader. If the file is located on the first sector of the specified partition, use the blocklist notation, **`+1`**, instead of the file name.

  The following is an example **`chainloader`** command:

  ```
  chainloader +1
  ```

- **`displaymem`** — Displays the current use of memory, based on information from the BIOS. This is useful to determine how much RAM a system has prior to booting it.
- **`initrd` *`</path/to/initrd>`*** — Enables users to specify an initial RAM disk to use when booting. An **`initrd`** is necessary when the kernel needs certain modules in order to boot properly, such as when the root partition is formatted with the ext3 file system.

  The following is an example **`initrd`** command:

  ```
  initrd /initrd-2.6.8-1.523.img
  ```

- **`install` *`<stage-1>`* *`<install-disk>`* *`<stage-2>`* `p` *`config-file`*** — Installs GRUB to the system MBR.
  - ***`<stage-1>`*** — Signifies a device, partition, and file where the first boot loader image can be found, such as **`(hd0,0)/grub/stage1`**.
  - ***`<install-disk>`*** — Specifies the disk where the stage 1 boot loader should be installed, such as **`(hd0)`**.
  - ***`<stage-2>`*** — Passes the stage 2 boot loader location to the stage 1 boot loader, such as **`(hd0,0)/grub/stage2`**.
  - **`p`** ***`<config-file>`*** — This option tells the **`install`** command to look for the menu configuration file specified by ***`<config-file>`***, such as **`(hd0,0)/grub/grub.conf`**.

> **Warning**
>
> The **`install`** command overwrites any information already located on the MBR.

- **`kernel` *`</path/to/kernel>`* *`<option-1>`* *`<option-N>`*** ... — Specifies the kernel file to load when booting the operating system. Replace *</path/to/kernel>* with an absolute path from the partition specified by the root command. Replace *<option-1>* with options for the Linux kernel, such as **`root=/dev/VolGroup00/LogVol00`** to specify the device on which the root partition for the system is located. Multiple options can be passed to the kernel in a space separated list.

  The following is an example **`kernel`** command:

  ```
  kernel /vmlinuz-2.6.8-1.523 ro root=/dev/VolGroup00/LogVol00
  ```

  The option in the previous example specifies that the root file system for Linux is located on the **`hda5`** partition.

- **root (*<device-type><device-number>*,*<partition>*)** — Configures the root partition for GRUB, such as **(hd0,0)**, and mounts the partition.

  The following is an example **root** command:

  ```
  root (hd0,0)
  ```

- **rootnoverify (*<device-type><device-number>*,*<partition>*)** — Configures the root partition for GRUB, just like the **root** command, but does not mount the partition.

Other commands are also available; type **help --all** for a full list of commands. For a description of all GRUB commands, refer to the documentation available online at http://www.gnu.org/software/grub/manual/.

9.7. GRUB Menu Configuration File

The configuration file (**/boot/grub/grub.conf**), which is used to create the list of operating systems to boot in GRUB's menu interface, essentially allows the user to select a pre-set group of commands to execute. The commands given in Section 9.6, “GRUB Commands” can be used, as well as some special commands that are only available in the configuration file.

9.7.1. Configuration File Structure

The GRUB menu interface configuration file is **/boot/grub/grub.conf**. The commands to set the global preferences for the menu interface are placed at the top of the file, followed by stanzas for each operating kernel or operating system listed in the menu.

The following is a very basic GRUB menu configuration file designed to boot either Red Hat Enterprise Linux or Microsoft Windows 2000:

```
default=0
timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Red Hat Enterprise Linux Server (2.6.18-2.el5PAE)
root (hd0,0)
kernel /boot/vmlinuz-2.6.18-2.el5PAE ro root=LABEL=/1 rhgb quiet
initrd /boot/initrd-2.6.18-2.el5PAE.img

# section to load Windows
title Windows
rootnoverify (hd0,0)
chainloader +1
```

This file configures GRUB to build a menu with Red Hat Enterprise Linux as the default operating system and sets it to autoboot after 10 seconds. Two sections are given, one for each operating system entry, with commands specific to the system disk partition table.

> **Note**
>
> Note that the default is specified as an integer. This refers to the first **title** line in the GRUB configuration file. For the **Windows** section to be set as the default in the previous example, change the **default=0** to **default=1**.

Configuring a GRUB menu configuration file to boot multiple operating systems is beyond the scope of this chapter. Consult Section 9.9, “Additional Resources” for a list of additional resources.

9.7.2. Configuration File Directives

The following are directives commonly used in the GRUB menu configuration file:

- **chainloader </*path/to/file*>** — Loads the specified file as a chain loader. Replace *</path/to/file>* with the absolute path to the chain loader. If the file is located on the first sector of the specified partition, use the blocklist notation, **+1**.
- **color <*normal-color*> <*selected-color*>** — Allows specific colors to be used in the menu, where two colors are configured as the foreground and background. Use simple color names such as **red/black**. For example:

  ```
  color red/black green/blue
  ```

- **default=<*integer*>** — Replace *<integer>* with the default entry title number to be loaded if the menu interface times out.
- **fallback=<*integer*>** — Replace *<integer>* with the entry title number to try if the first attempt fails.
- **hiddenmenu** — Prevents the GRUB menu interface from being displayed, loading the **default** entry when the **timeout** period expires. The user can see the standard GRUB menu by pressing the **Esc** key.
- **initrd </*path/to/initrd*>** — Enables users to specify an initial RAM disk to use when booting. Replace *</path/to/initrd>* with the absolute path to the initial RAM disk.
- **kernel </*path/to/kernel*> <*option-1*> <*option-N*>** — Specifies the kernel file to load when booting the operating system. Replace *</path/to/kernel>* with an absolute path from the partition specified by the root directive. Multiple options can be passed to the kernel when it is loaded.
- **password=<*password*>** — Prevents a user who does not know the password from editing the entries for this menu option.

  Optionally, it is possible to specify an alternate menu configuration file after the **password=<*password*>** directive. In this case, GRUB restarts the second stage boot loader and uses the specified alternate configuration file to build the menu. If an alternate menu configuration file is left out of the command, a user who knows the password is allowed to edit the current configuration file.

  For more information about securing GRUB, see the chapter titled *Workstation Security* in the Red Hat Enterprise Linux Deployment Guide.

- **root (<*device-type*><*device-number*>,<*partition*>)** — Configures the root partition for GRUB, such as **(hd0,0)**, and mounts the partition.
- **rootnoverify (<*device-type*><*device-number*>,<*partition*>)** — Configures the root partition for GRUB, just like the **root** command, but does not mount the partition.
- **timeout=<*integer*>** — Specifies the interval, in seconds, that GRUB waits before loading the entry designated in the **default** command.

- **splashimage=<*path-to-image*>** — Specifies the location of the splash screen image to be used when GRUB boots.
- **title *group-title*** — Specifies a title to be used with a particular group of commands used to load a kernel or operating system.

To add human-readable comments to the menu configuration file, begin the line with the hash mark character (**#**).

9.8. Changing Runlevels at Boot Time

Under Red Hat Enterprise Linux, it is possible to change the default runlevel at boot time.

To change the runlevel of a single boot session, use the following instructions:

- When the GRUB menu bypass screen appears at boot time, press any key to enter the GRUB menu (within the first three seconds).
- Press the **a** key to append to the **kernel** command.
- Add **<space><*runlevel*>** at the end of the boot options line to boot to the desired runlevel. For example, the following entry would initiate a boot process into runlevel 3:

```
grub append> ro root=/dev/VolGroup00/LogVol00 rhgb quiet 3
```

9.9. Additional Resources

This chapter is only intended as an introduction to GRUB. Consult the following resources to discover more about how GRUB works.

9.9.1. Installed Documentation

- **/usr/share/doc/grub-<*version-number*>/** — This directory contains good information about using and configuring GRUB, where *<version-number>* corresponds to the version of the GRUB package installed.
- **info grub** — The GRUB info page contains a tutorial, a user reference manual, a programmer reference manual, and a FAQ document about GRUB and its usage.

9.9.2. Useful Websites

- http://www.gnu.org/software/grub/ — The home page of the GNU GRUB project. This site contains information concerning the state of GRUB development and an FAQ.
- http://kbase.redhat.com/faq/FAQ_43_4053.shtm — Details booting operating systems other than Linux.
- http://www.linuxgazette.com/issue64/kohli.html — An introductory article discussing the configuration of GRUB on a system from scratch, including an overview of GRUB command line options.

9.9.3. Related Books

- Red Hat Enterprise Linux Deployment Guide; Red Hat, Inc. — The *Workstation Security* chapter explains, in a concise manner, how to secure the GRUB boot loader.

[4] For more on the system BIOS and the MBR, refer to Section 33.2.1, “The BIOS”.

Chapter 10. Additional Resources about Itanium and Linux

Other reference materials, related to running Red Hat Enterprise Linux on an Itanium system, are available on the Web. A few of the available resources are as follows:

- / — The Intel website on the Itanium Processor
- http://developer.intel.com/technology/efi/index.htm?iid=sr+efi — The Intel website for the Extensible Firmware Interface (EFI)
- http://www.itanium.com/business/bss/products/server/itanium2/index.htm — The Intel website on the Itanium 2 processor

Part II. IBM POWER Architecture - Installation and Booting

The *Red Hat Enterprise Linux Installation Guide* for IBM POWER systems discusses the installation of Red Hat Enterprise Linux and some basic post-installation troubleshooting. Advanced installation options are covered in the second part of this manual.

Chapter 11. Steps to Get You Started

11.1. Upgrade or Install?

For information to help you determine whether to perform an upgrade or an installation refer to Chapter 24, *Upgrading Your Current System*.

11.2. Preparation for IBM eServer System p and System i

The IBM eServer System p and IBM eServer System i systems introduce many options for partitioning, virtual or native devices, and consoles. Both versions of the system use the same kernel and have many of the same options available, depending on the system configuration.

If you are using a non-partitioned System p system, you do not need any pre-installation setup. For systems using the HVSI serial console, hook up your console to the T2 serial port.

If using a partitioned system, whether IBM System p or IBM System i the steps to create the partition and start the installation are largely the same. You should create the partition at the HMC and assign some CPU and memory resources, as well as SCSI and Ethernet resources, which can be either virtual or native. The HMC create partition wizard steps you through the creation.

For more information on creating the partition, refer to IBM's Infocenter article on *Configuring Linux logical partitions* available online at: http://publib.boulder.ibm.com/infocenter/eserver/v1r3s/index.jsp?topic=/iphbi/iphbikickoff.htm

If you are using virtual SCSI resources, rather than native SCSI, you must configure a 'link' to the virtual SCSI serving partition, and then configure the virtual SCSI serving partition itself. You create a 'link' between the virtual SCSI client and server slots using the HMC. You can configure a virtual SCSI server on either AIX or i5/OS, depending on which model and options you have.

For more information on using virtual devices, including IBM Redbooks and other online resources see: http://publib.boulder.ibm.com/infocenter/eserver/v1r3s/index.jsp?topic=/iphbi/iphbirelated.htm

Additional information on virtualization eServer i5 can be found in the IBM Redbook SG24-6388-01, *Implementing POWER Linux on IBM System i Platform*. This can be accessed at: http://www.redbooks.ibm.com/redpieces/abstracts/sg246388.html?Open

Once you have your system configured, you need to Activate from the HMC or power it on. Depending on what type of install you are doing, you may need to configure SMS to correctly boot the system into the installation program.

11.3. Do You Have Enough Disk Space?

Nearly every modern-day operating system (OS) uses *disk partitions*, and Red Hat Enterprise Linux is no exception. When you install Red Hat Enterprise Linux, you may have to work with disk partitions. If you have not worked with disk partitions before (or need a quick review of the basic concepts), refer to Chapter 26, *An Introduction to Disk Partitions* before proceeding.

The disk space used by Red Hat Enterprise Linux must be separate from the disk space used by other OSes you may have installed on your system.

Before you start the installation process, you must

- have enough *unpartitioned* [5] disk space for the installation of Red Hat Enterprise Linux, or

- have one or more partitions that may be deleted, thereby freeing up enough disk space to install Red Hat Enterprise Linux.

To gain a better sense of how much space you really need, refer to the recommended partitioning sizes discussed in Section 12.19.4, “Recommended Partitioning Scheme”.

11.4. Can You Install Using the CD-ROM or DVD?

Installing from a CD-ROM or DVD requires that you have purchased a Red Hat Enterprise Linux product, you have a Red Hat Enterprise Linux 5.11 CD-ROM or DVD, and you have a DVD/CD-ROM drive on a system that supports booting from it.

11.5. Preparing for a Network Installation

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

The Red Hat Enterprise Linux installation media must be available for either a network installation (via NFS, FTP, or HTTP) or installation via local storage. Use the following steps if you are performing an NFS, FTP, or HTTP installation.

The NFS, FTP, or HTTP server to be used for installation over the network must be a separate machine which can provide the complete contents of the installation DVD-ROM or the installation CD-ROMs.

> **Note**
>
> The Red Hat Enterprise Linux installation program has the ability to test the integrity of the installation media. It works with the CD / DVD, hard drive ISO, and NFS ISO installation methods. Red Hat recommends that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned CDs). To use this test, type the following command at the **`yaboot:`** prompt:
>
> ```
> linux mediacheck
> ```

> **Note**
>
> In the following examples, the directory on the installation staging server that will contain the installation files will be specified as ***`/location/of/disk/space`***. The directory that will be made publicly available via FTP, NFS, or HTTP will be specified as */publicly/available/directory*. For example, ***`/location/of/disk/space`*** may be a directory you create called **`/var/isos`**. ***`/publicly/available/directory`*** might be **`/var/www/html/rhel5`**, for an HTTP install.

To copy the files from the installation DVD or CD-ROMs to a Linux machine which acts as an installation staging server, perform the following steps:

- Create an iso image from the installation disk(s) using the following command (for DVDs):

  ```
  dd if=/dev/dvd of=/location/of/disk/space/RHEL5.iso
  ```

where *dvd* refers to your DVD drive device.

11.5.1. Preparing for FTP and HTTP installation

If you are setting up an installation tree for NFS, FTP, or HTTP installations, you must copy the **RELEASE-NOTES** files and all files from the **RedHat** directory on all operating systems ISO images. On Linux and UNIX systems, the following process will properly configure the target directory on your server (repeat for each CD-ROM/ISO image):

1. Insert CD-ROM or DVD-ROM.
2. **mount /media/cdrom**
3. If you are installing the Server variant, run **cp -a /media/cdrom/Server *<target-directory>***

   If you are installing the Client variant, run **cp -a /media/cdrom/Client *<target-directory>***
4. **cp /media/cdrom/RELEASE-NOTES* *<target-directory>*** (Installation CD 1 or DVD only)
5. **cp /media/cdrom/images *<target-directory>*** (Installation CD 1 or DVD only)
6. **umount /media/cdrom**

(Where ***<target-directory>*** represents the path to the directory to contain the installation tree.)

> **Note**
>
> Do *not* copy the Supplementary ISO image, or any of the layered product ISO images, as this will overwrite files necessary for Anaconda's proper operation.
>
> These ISO images must be installed *after* Red Hat Enterprise Linux has been installed.

Next make sure that the ***/publicly/available/directory*** directory is shared via FTP or HTTP, and verify client access. You can check to see whether the directory is accessible from the server itself, and then from another machine on the same subnet that you will be installing to.

11.5.2. Preparing for an NFS install

For NFS installation it is not necessary to mount the iso image. It is sufficient to make the iso image itself available via NFS. You can do this by moving the iso image or images to the NFS exported directory:

- For DVD:

  mv */location/of/disk/space/*RHEL5.iso */publicly/available/directory/*

- For CDROMs:

  mv */location/of/disk/space/*disk*.iso */publicly/available/directory/*

Ensure that the ***/publicly/available/directory*** directory is exported via NFS via an entry in **/etc/exports**.

To export to a specific system:

```
/publicly/available/directory client.ip.address
```

To export to all systems use an entry such as:

```
/publicly/available/directory *
```

Start the NFS daemon (on a Red Hat Enterprise Linux system, use **/sbin/service nfs start**). If NFS is already running, reload the configuration file (on a Red Hat Enterprise Linux system use **/sbin/service nfs reload**).

Be sure to test the NFS share following the directions in the Red Hat Enterprise Linux Deployment Guide.

11.6. Preparing for a Hard Drive Installation

> **Note**
>
> Hard drive installations only work from ext2, ext3, or FAT file systems. If you have a file system other than those listed here, such as reiserfs, you will not be able to perform a hard drive installation.

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

Hard drive installations require the use of the ISO (or DVD/CD-ROM) images. An ISO image is a file containing an exact copy of a DVD/CD-ROM image. After placing the required ISO images (the binary Red Hat Enterprise Linux DVD/CD-ROMs) in a directory, choose to install from the hard drive. You can then point the installation program at that directory to perform the installation.

To prepare your system for a hard drive installation, you must set the system up in one of the following ways:

- Using a set of CD-ROMs, or a DVD — Create ISO image files from each installation CD-ROM, or from the DVD. For each CD-ROM (once for the DVD), execute the following command on a Linux system:

  ```
  dd if=/dev/cdrom of=/tmp/file-name.iso
  ```

- Using ISO images — transfer these images to the system to be installed.

  Verifying that ISO images are intact before you attempt an installation, helps to avoid problems. To verify the ISO images are intact prior to performing an installation, use an **md5sum** program (many **md5sum** programs are available for various operating systems). An **md5sum** program should be available on the same Linux machine as the ISO images.

Additionally, if a file called **updates.img** exists in the location from which you install, it is used for updates to **anaconda**, the installation program. Refer to the file **install-methods.txt** in the **anaconda** RPM package for detailed information on the various ways to install Red Hat Enterprise Linux, as well as how to apply the installation program updates.

[5] Unpartitioned disk space means that available disk space on the hard drive(s) you are installing to has not been divided into sections for data. When you partition a disk, each partition behaves like a separate disk drive.

Chapter 12. Installing on IBM System i and IBM System p systems

This chapter explains how to perform a Red Hat Enterprise Linux installation from the DVD/CD-ROM, using the graphical, mouse-based installation program. The following topics are discussed:

- Becoming familiar with the installation program's user interface
- Starting the installation program
- Selecting an installation method
- Configuration steps during the installation (language, keyboard, mouse, partitioning, etc.)
- Finishing the installation

12.1. The Graphical Installation Program User Interface

If you have used a *graphical user interface (GUI)* before, you are already familiar with this process; use your mouse to navigate the screens, click buttons, or enter text fields.

You can also navigate through the installation using the keyboard. The **Tab** key allows you to move around the screen, the Up and Down arrow keys to scroll through lists, **+** and **-** keys expand and collapse lists, while **Space** and **Enter** selects or removes from selection a highlighted item. You can also use the **Alt**+***X*** key command combination as a way of clicking on buttons or making other screen selections, where ***X*** is replaced with any underlined letter appearing within that screen.

If you would like to use a graphical installation with a system that does not have that capability, such as a partitioned system, you can use VNC or display forwarding. Both the VNC and display forwarding options require an active network during the installation and the use of boot time arguments. For more information on available boot time options, refer to Chapter 15, *Additional Boot Options for IBM Power Systems*

Note

If you do not wish to use the GUI installation program, the text mode installation program is also available. To start the text mode installation program, use the following command at the **yaboot:** prompt:

```
linux text
```

Refer to Section 12.5, “The Text Mode Installation Program User Interface” for a brief overview of text mode installation instructions.

It is highly recommended that installs be performed using the GUI installation program. The GUI installation program offers the full functionality of the Red Hat Enterprise Linux installation program, including LVM configuration which is not available during a text mode installation.

Users who must use the text mode installation program can follow the GUI installation instructions and obtain all needed information.

12.2. Booting the IBM System i or IBM System p Installation Program

To boot an IBM System i or IBM System p system from a CD-ROM, you must specify the install boot device in the **`System Management Services`** (SMS) menu.

To enter the **`System Management Services`** GUI, press the **`1`** key during the boot process when you hear the chime sound. This brings up a graphical interface similar to the one described in this section.

On a text console, press **`1`** when the self test is displaying the banner along with the tested components:

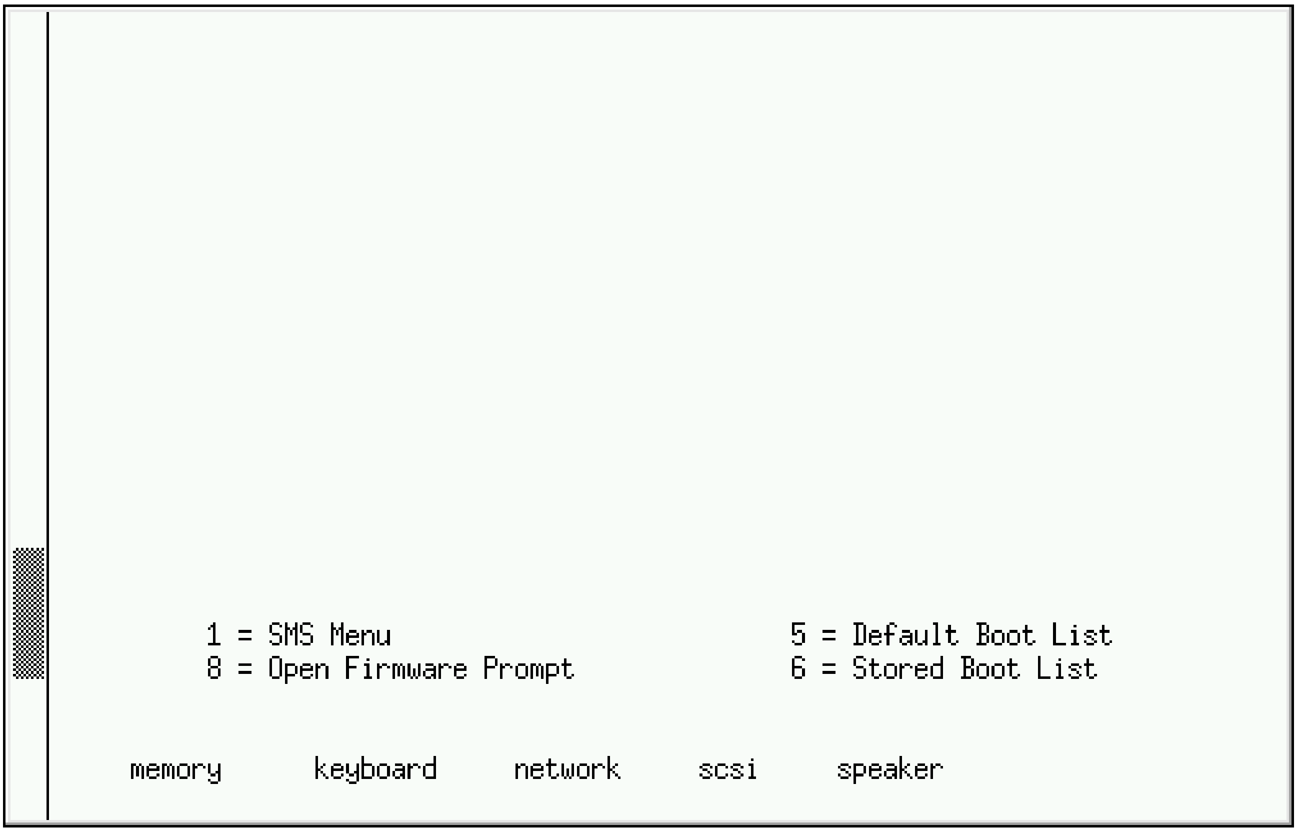

Figure 12.1. SMS console

Once in the SMS menu, select the option for **Select Boot Options**. In that menu, specify **`Select Install or Boot a Device`**. There, select **`CD/DVD`**, and then the bus type (in most cases SCSI). If you are uncertain, you can select to view all devices. This scans all available buses for boot devices, including network adapters and hard drives.

Finally, select the device containing the installation CD. YABOOT is loaded from this device and you are presented with a **`boot:`** prompt. Press **`Enter`** or wait for the timeout to expire for the installation to begin.

If you are booting via the network, use the **`images/netboot/ppc64.img`** file on CD #1.

12.3. A Note about Linux Virtual Consoles

This information only applies to users of non-partitioned System p systems using a video card as their console. Users of partitioned System i and System p systems should skip to Section 12.4, “Using the HMC vterm”.

The Red Hat Enterprise Linux installation program offers more than the dialog boxes of the installation process. Several kinds of diagnostic messages are available to you, as well as a way to enter commands from a shell prompt. The installation program displays these messages on five *virtual consoles*, among which you can switch using a single keystroke combination.

A virtual console is a shell prompt in a non-graphical environment, accessed from the physical machine, not remotely. Multiple virtual consoles can be accessed simultaneously.

These virtual consoles can be helpful if you encounter a problem while installing Red Hat Enterprise Linux. Messages displayed on the installation or system consoles can help pinpoint a problem. Refer to Table 12.1, “Console, Keystrokes, and Contents” for a listing of the virtual consoles, keystrokes used to switch to them, and their contents.

Generally, there is no reason to leave the default console (virtual console #6) for graphical installations unless you are attempting to diagnose installation problems.

Table 12.1. Console, Keystrokes, and Contents

console	keystrokes	contents
1	**ctrl**+**alt**+**f1**	installation dialog
2	**ctrl**+**alt**+**f2**	shell prompt
3	**ctrl**+**alt**+**f3**	install log (messages from installation program)
4	**ctrl**+**alt**+**f4**	system-related messages
5	**ctrl**+**alt**+**f5**	other messages
6	**ctrl**+**alt**+**f6**	x graphical display

12.4. Using the HMC vterm

The HMC vterm is the console for any partitioned IBM System p or IBM System i system. This is opened by right clicking on the partition on the HMC, and then selecting **`Open Terminal Window`**. Only a single vterm can be connected to the console at one time and there is no console access for partitioned system besides the vterm. This often is referred to as a 'virtual console', but is different from the virtual consoles in Section 12.3, “A Note about Linux Virtual Consoles” .

12.5. The Text Mode Installation Program User Interface

The Red Hat Enterprise Linux text mode installation program uses a screen-based interface that includes most of the on-screen *widgets* commonly found on graphical user interfaces. Figure 12.2, “Installation Program Widgets as seen in **`Boot Loader Configuration`**”, and Figure 12.3, “Installation Program Widgets as seen in Disk Druid”, illustrate the screens that appear during the installation process.

> **Note**
>
> While text mode installations are not explicitly documented, those using the text mode installation program can easily follow the GUI installation instructions. One thing to note is that manipulation of LVM (Logical Volume Management) disk volumes is only possible in graphical mode. In text mode it is only possible to view and accept the default LVM setup.

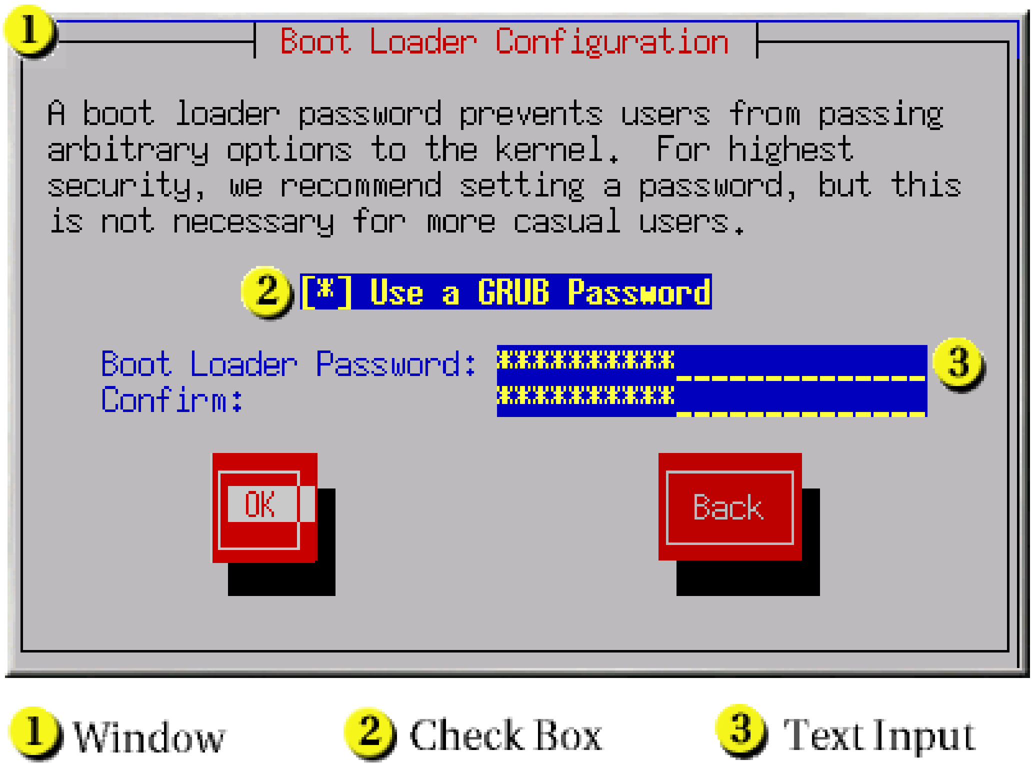

Figure 12.2. Installation Program Widgets as seen in Boot Loader Configuration

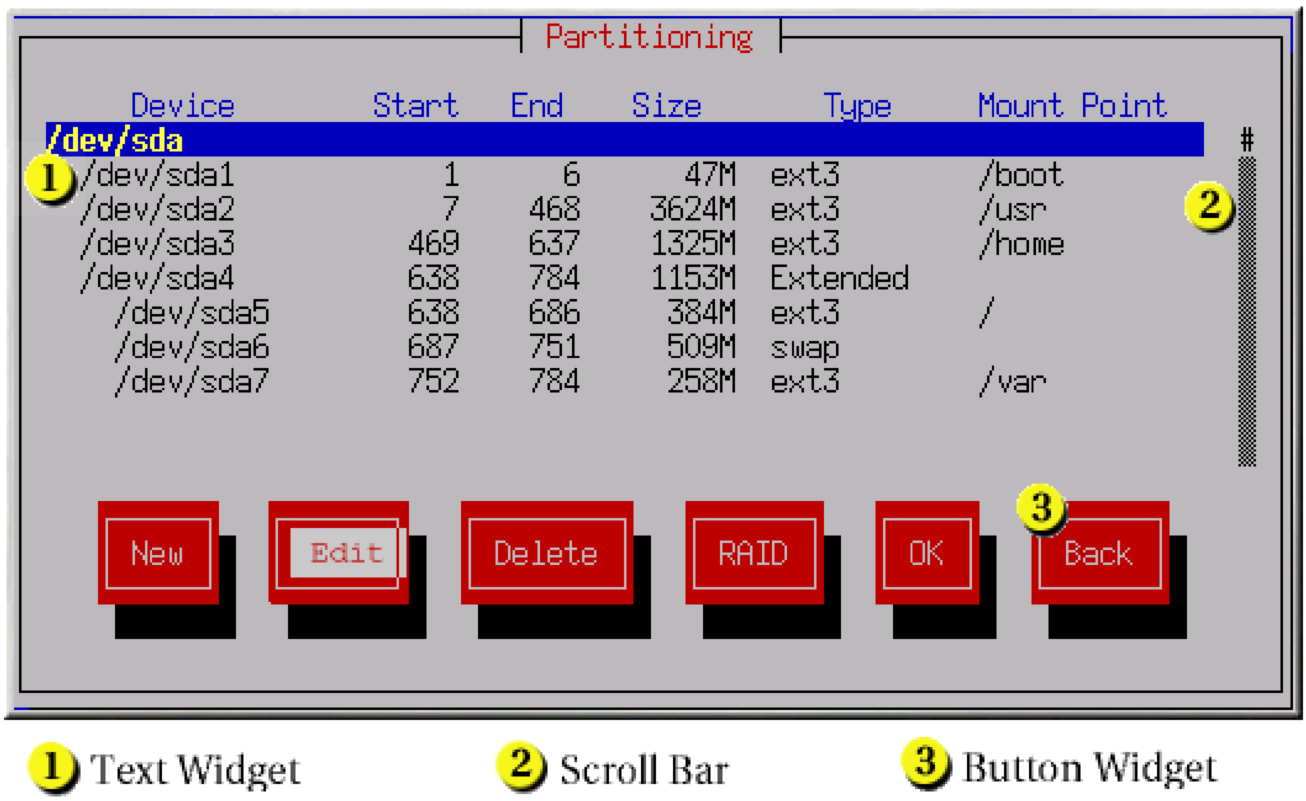

Figure 12.3. Installation Program Widgets as seen in Disk Druid

Here is a list of the most important widgets shown in Figure 12.2, “Installation Program Widgets as seen in **Boot Loader Configuration**” and Figure 12.3, “Installation Program Widgets as seen in Disk Druid”:

- Window — Windows (usually referred to as *dialogs* in this manual) appear on your screen throughout the installation process. At times, one window may overlay another; in these cases, you can only interact with the window on top. When you are finished in that window, it disappears, allowing you to continue working in the window underneath.

- Checkbox — Checkboxes allow you to select or deselect a feature. The box displays either an asterisk (selected) or a space (unselected). When the cursor is within a checkbox, press **Space** to select or deselect a feature.

- Text Input — Text input lines are regions where you can enter information required by the installation program. When the cursor rests on a text input line, you may enter and/or edit information on that line.

- Text Widget — Text widgets are regions of the screen for the display of text. At times, text widgets may also contain other widgets, such as checkboxes. If a text widget contains more information than can be displayed in the space reserved for it, a scroll bar appears; if you position the cursor within the text widget, you can then use the **Up** and **Down** arrow keys to scroll through all the information available. Your current position is shown on the scroll bar by a # character, which moves up and down the scroll bar as you scroll.

- Scroll Bar — Scroll bars appear on the side or bottom of a window to control which part of a list or document is currently in the window's frame. The scroll bar makes it easy to move to any part of a file.

- Button Widget — Button widgets are the primary method of interacting with the installation program. You progress through the windows of the installation program by navigating these buttons, using the **Tab** and **Enter** keys. Buttons can be selected when they are highlighted.

- Cursor — Although not a widget, the cursor is used to select (and interact with) a particular widget. As the cursor is moved from widget to widget, it may cause the widget to change color, or the cursor itself may only appear positioned in or next to the widget. In Figure 12.2, “Installation Program Widgets as seen in **Boot Loader Configuration**”, the cursor is positioned on the **OK** button. Figure 12.3, “Installation Program Widgets as seen in Disk Druid”, shows the cursor on the **Edit** button.

12.5.1. Using the Keyboard to Navigate

Navigation through the installation dialogs is performed through a simple set of keystrokes. To move the cursor, use the **Left**, **Right**, **Up**, and **Down** arrow keys. Use **Tab**, and **Shift**-**Tab** to cycle forward or backward through each widget on the screen. Along the bottom, most screens display a summary of available cursor positioning keys.

To "press" a button, position the cursor over the button (using **Tab**, for example) and press **Space** or **Enter**. To select an item from a list of items, move the cursor to the item you wish to select and press **Enter**. To select an item with a checkbox, move the cursor to the checkbox and press **Space** to select an item. To deselect, press **Space** a second time.

Pressing **F12** accepts the current values and proceeds to the next dialog; it is equivalent to pressing the **OK** button.

Warning

Unless a dialog box is waiting for your input, do not press any keys during the installation process (doing so may result in unpredictable behavior).

12.6. Beginning Installation

12.6.1. Installing from DVD/CD-ROM

To install Red Hat Enterprise Linux from a DVD/CD-ROM, place the DVD or CD #1 in your DVD/CD-ROM drive and boot your system from the DVD/CD-ROM.

The installation program then probes your system and attempts to identify your CD-ROM drive. It starts by looking for an IDE (also known as an ATAPI) CD-ROM drive.

If your CD-ROM drive is not detected, and it is a SCSI CD-ROM, the installation program prompts you to choose a SCSI driver. Choose the driver that most closely resembles your adapter. You may specify options for the driver if necessary; however, most drivers detect your SCSI adapter automatically.

If the DVD/CD-ROM drive is found and the driver loaded, the installer will present you with the option to perform a media check on the DVD/CD-ROM. This will take some time, and you may opt to skip over this step. However, if you later encounter problems with the installer, you should reboot and perform the media check before calling for support. From the media check dialog, continue to the next stage of the installation process (refer to Section 12.12, “Welcome to Red Hat Enterprise Linux”).

12.7. Installing from a Hard Drive

The **Select Partition** screen applies only if you are installing from a disk partition (that is, if you used the **askmethod** boot options and selected **Hard Drive** in the **Installation Method** dialog). This dialog allows you to name the disk partition and directory from which you are installing Red Hat Enterprise Linux. If you used the **repo=hd** boot option, you already specified a partition.

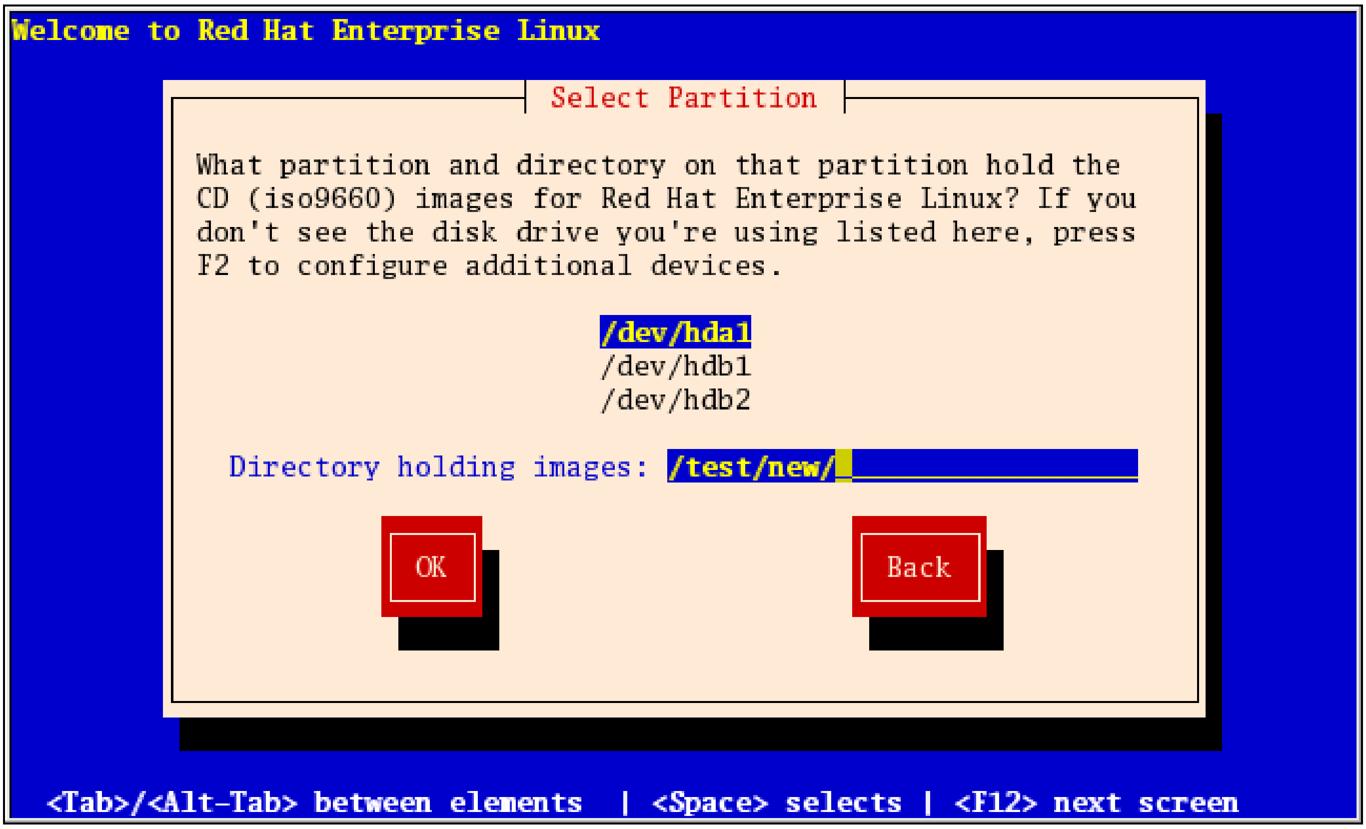

Figure 12.4. Selecting Partition Dialog for Hard Drive Installation

Enter the device name of the partition containing the Red Hat Enterprise Linux ISO images. This partition must be formatted with a ext2 or vfat filesystem, and cannot be a logical volume. There is also a field labeled **`Directory holding images`**.

If the ISO images are in the root (top-level) directory of a partition, enter a `/`. If the ISO images are located in a subdirectory of a mounted partition, enter the name of the directory holding the ISO images within that partition. For example, if the partition on which the ISO images is normally mounted as **`/home/`**, and the images are in **`/home/new/`**, you would enter **`/new/`**.

After you have identified the disk partition, the **`Welcome`** dialog appears.

12.8. Performing a Network Installation

If you are performing a network installation and booted with the **`askmethod`** boot option, the **`Configure TCP/IP`** dialog appears. This dialog asks for your IP and other network addresses. You can choose to configure the IP address and Netmask of the device via DHCP or manually. If manually, you have the option to enter IPv4 and/or IPv6 information. Enter the IP address you are using during installation and press **`Enter`**. Note that you need to supply IPv4 information if you wish to perform an NFS installation.

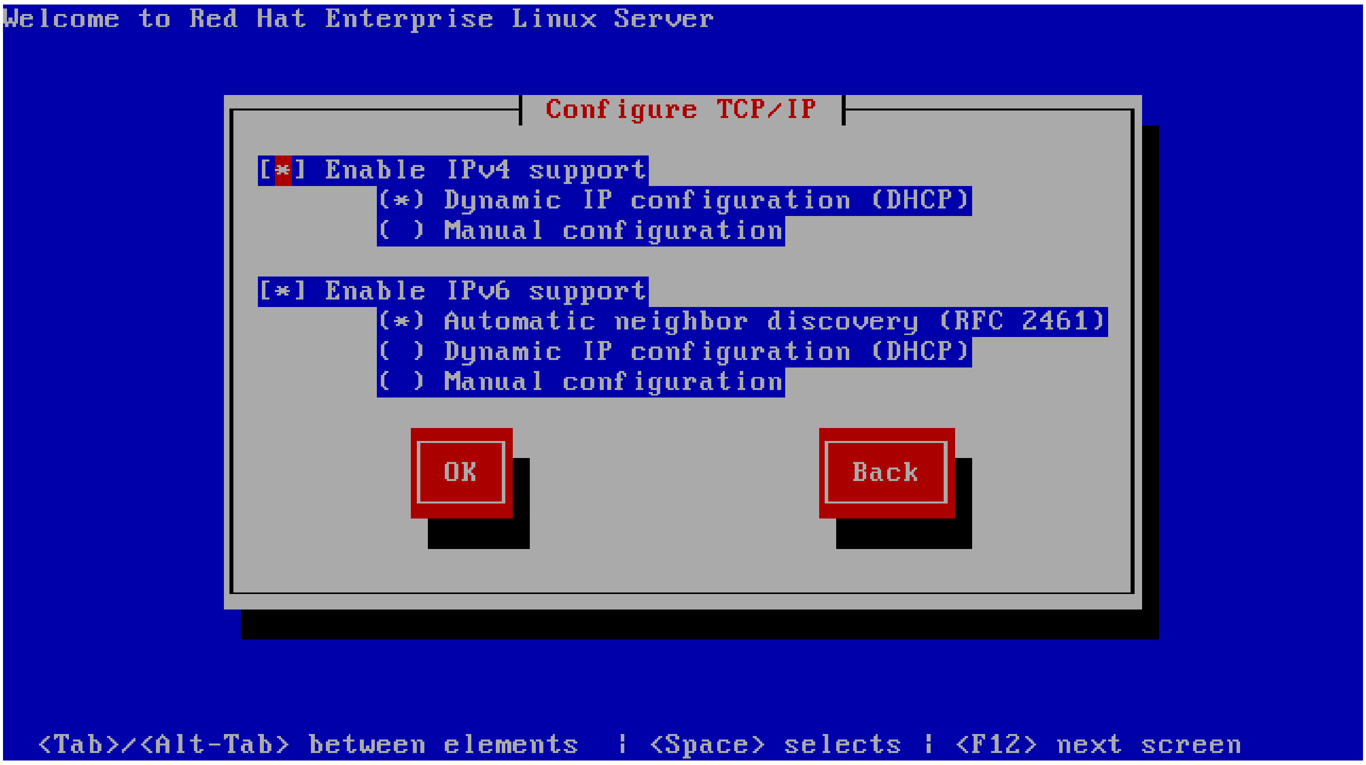

Figure 12.5. TCP/IP Configuration

12.9. Installing via NFS

The NFS dialog applies only if you are installing from an NFS server (if you selected **NFS Image** in the **`Installation Method`** dialog).

Enter the domain name or IP address of your NFS server. For example, if you are installing from a host named **`eastcoast`** in the domain **`example.com`**, enter **`eastcoast.example.com`** in the **`NFS Server`** field.

Next, enter the name of the exported directory. If you followed the setup described in Section 11.5, "Preparing for a Network Installation", you would enter the directory ***`/export/directory/`*** which contains the ***`variant/`*** directory.

If the NFS server is exporting a mirror of the Red Hat Enterprise Linux installation tree, enter the directory which contains the root of the installation tree. You will enter an Installation Key later on in the process which will determine which subdirectories are used to install from. If everything was specified properly, a message appears indicating that the installation program for Red Hat Enterprise Linux is running.

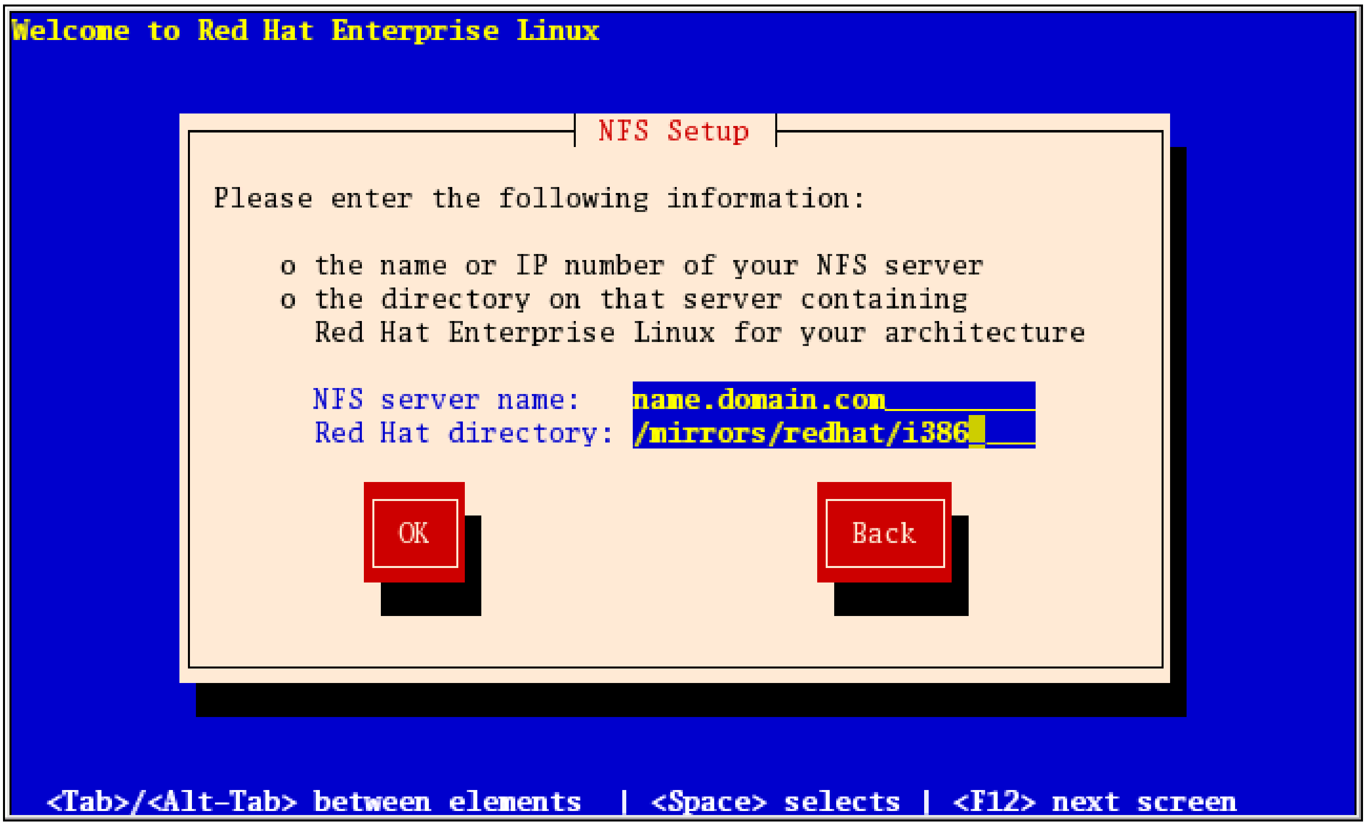

Figure 12.6. NFS Setup Dialog

If the NFS server is exporting the ISO images of the Red Hat Enterprise Linux CD-ROMs, enter the directory which contains the ISO images.

Next, the **`Welcome`** dialog appears.

12.10. Installing via FTP

The FTP dialog applies only if you are installing from an FTP server (that is, if you used the **`askmethod`** boot options and selected **FTP** in the **`Installation Method`** dialog). This dialog allows you to identify the FTP server from which you are installing Red Hat Enterprise Linux. If you used the **`repo=ftp`** boot option, you already specified a server and path.

Figure 12.7. FTP Setup Dialog

Enter the name or IP address of the FTP site you are installing from, and the name of the directory containing the ***variant*/** directory for your architecture. For example, if the FTP site contains the directory **/mirrors/redhat/*arch*/*variant*;/**, enter **/mirrors/redhat/*arch*/** (where *arch* is replaced with the architecture type of your system, such as i386, ia64, ppc, or s390x, and *variant* is the variant that you are installing, such as Client, Server, Workstation, etc.). If everything was specified properly, a message box appears indicating that files are being retrieved from the server.

Next, the **Welcome** dialog appears.

Note

You can save disk space by using the ISO images you have already copied to the server. To accomplish this, install Red Hat Enterprise Linux using ISO images without copying them into a single tree by loopback mounting them. For each ISO image:

mkdir disc*X*

mount -o loop RHEL5-disc*X*.iso disc*X*

Replace **X** with the corresponding disc number.

12.11. Installing via HTTP

The HTTP dialog applies only if you are installing from an HTTP server (that is, if you used the **askmethod** boot option and selected **HTTP** in the **Installation Method** dialog). This dialog prompts you for information about the HTTP server from which you are installing Red Hat Enterprise Linux. If you used the **repo=http** boot option, you already specified a server and path.

Enter the name or IP address of the HTTP site you are installing from, and the name of the directory containing the ***variant/*** directory for your architecture. For example, if the HTTP site contains the directory **/mirrors/redhat/*arch*/*variant*/**, enter **/mirrors/redhat/*arch*/** (where *arch* is replaced with the architecture type of your system, such as i386, ia64, ppc, or s390x, and *variant* is the variant that you are installing, such as Client, Server, Workstation, etc.). If everything was specified properly, a message box appears indicating that files are being retrieved from the server.

Figure 12.8. HTTP Setup Dialog

Next, the **Welcome** dialog appears.

Note

You can save disk space by using the ISO images you have already copied to the server. To accomplish this, install Red Hat Enterprise Linux using ISO images without copying them into a single tree by loopback mounting them. For each ISO image:

mkdir disc*X*

mount -o loop RHEL5-disc*X*.iso disc*X*

Replace **X** with the corresponding disc number.

12.12. Welcome to Red Hat Enterprise Linux

The **Welcome** screen does not prompt you for any input. From this screen you can access the Release Notes for Red Hat Enterprise Linux 5.11 by clicking on the Release Notes button.

Click on the **Next** button to continue.

12.13. Language Selection

Using your mouse, select a language to use for the installation (refer to Figure 12.9, “Language Selection”).

The language you select here will become the default language for the operating system once it is installed. Selecting the appropriate language also helps target your time zone configuration later in the installation. The installation program tries to define the appropriate time zone based on what you specify on this screen.

Figure 12.9. Language Selection

Once you select the appropriate language, click **Next** to continue.

12.14. Keyboard Configuration

Using your mouse, select the correct layout type (for example, U.S. English) for the keyboard you would prefer to use for the installation and as the system default (refer to Figure 12.10, “Keyboard Configuration”).

Once you have made your selection, click **Next** to continue.

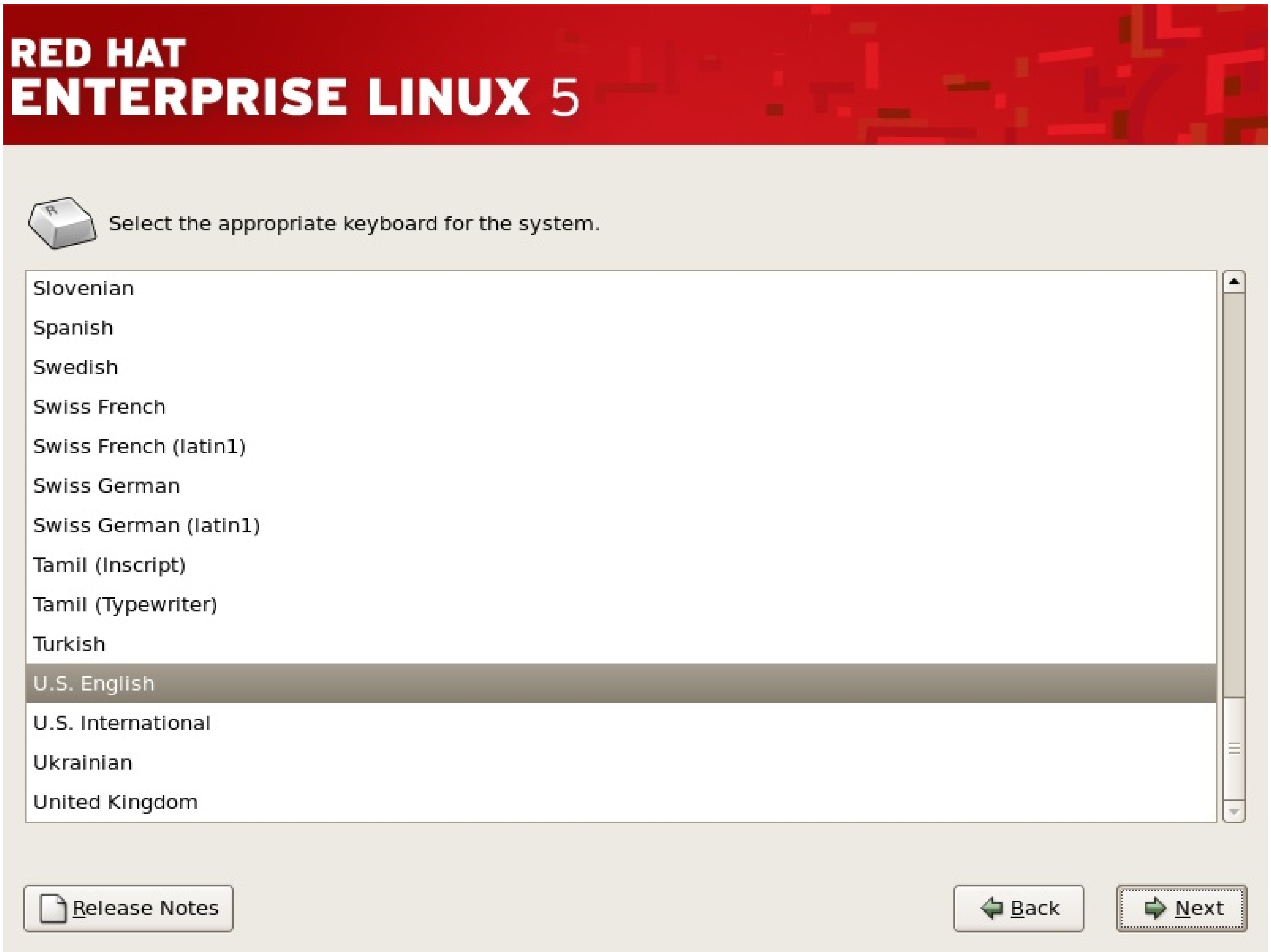

Figure 12.10. Keyboard Configuration

> **Note**
>
> To change your keyboard layout type after you have completed the installation, use the **Keyboard Configuration Tool**.
>
> Type the **`system-config-keyboard`** command in a shell prompt to launch the **Keyboard Configuration Tool**. If you are not root, it prompts you for the root password to continue.

12.15. Enter the Installation Number

Enter your Installation Number (refer to Figure 12.11, "Installation Number"). This number will determine the package selection set that is available to the installer. If you choose to skip entering the installation number you will be presented with a basic selection of packages to install later on.

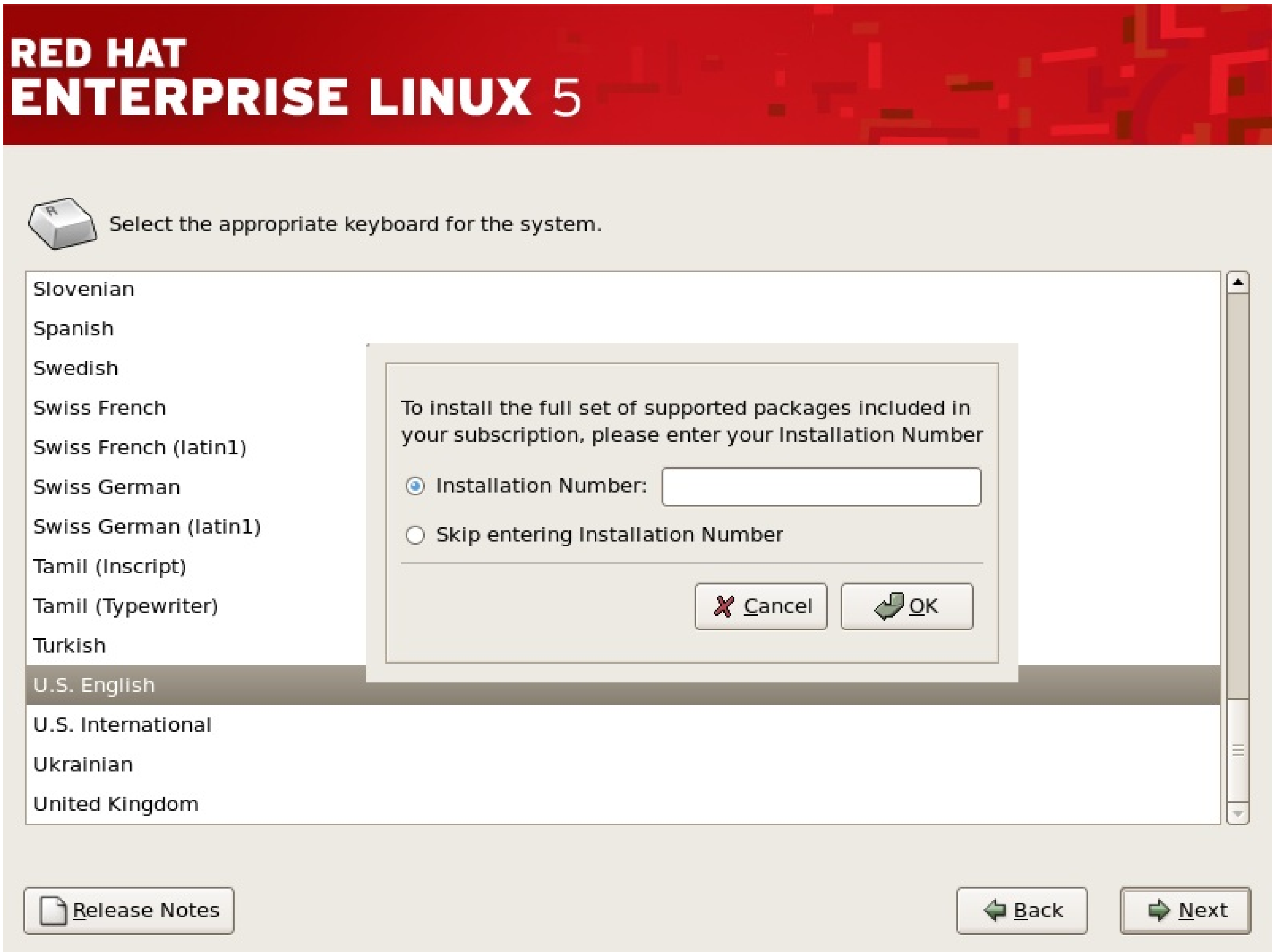

Figure 12.11. Installation Number

12.16. Disk Partitioning Setup

Partitioning allows you to divide your hard drive into isolated sections, where each section behaves as its own hard drive. Partitioning is particularly useful if you run multiple operating systems. If you are not sure how you want your system to be partitioned, read Chapter 26, *An Introduction to Disk Partitions* for more information.

On this screen you can choose to create the default layout or choose to manual partition using the 'Create custom layout' option of **Disk Druid**.

The first three options allow you to perform an automated installation without having to partition your drive(s) yourself. If you do not feel comfortable with partitioning your system, it is recommended that you *do not* choose to create a custom layout and instead let the installation program partition for you.

You can configure an iSCSI target for installation, or disable a dmraid device from this screen by clicking on the 'Advanced storage configuration' button. For more information refer to Section 12.17, “ Advanced Storage Options ”.

Warning

The **Update Agent** downloads updated packages to **/var/cache/yum/** by default. If you partition the system manually, and create a separate **/var/** partition, be sure to create the partition large enough (3.0 GB or more) to download package updates.

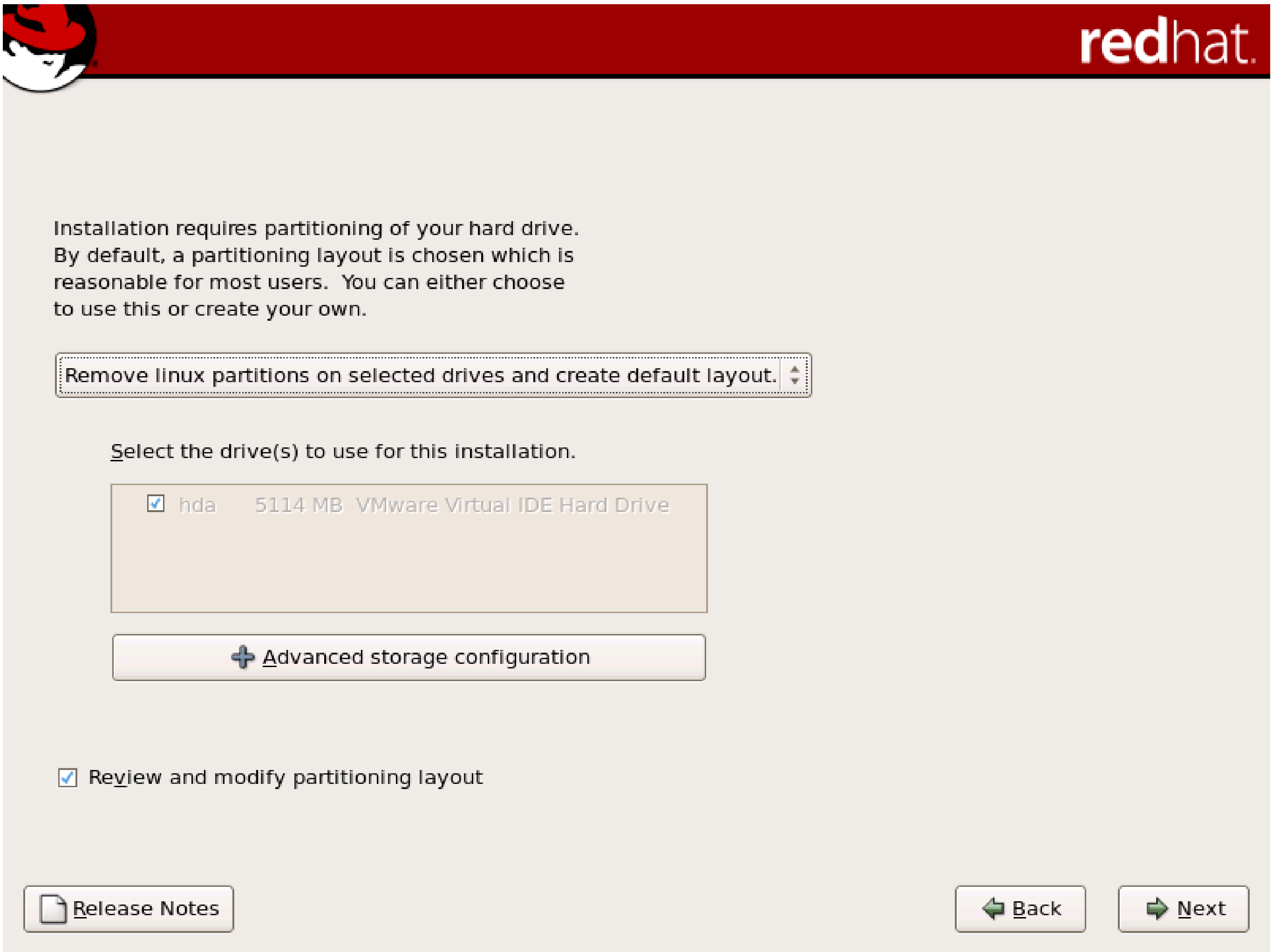

Figure 12.12. Disk Partitioning Setup

If you choose to create a custom layout using **Disk Druid**, refer to Section 12.19, “Partitioning Your System”.

Warning

If you receive an error after the **`Disk Partitioning Setup`** phase of the installation saying something similar to:

"The partition table on device hda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive."

No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

Important — multipath devices

To install Red Hat Enterprise Linux on a network device accessible through multiple paths, deselect all local storage in the **`Select the drive(s) to use for this installation`** window, and select a device labeled **`mapper/mpath`** instead.

Note that migrating the root file system of an existing Red Hat Enterprise Linux installation from single path storage to multipath storage is not supported. You must perform a new installation to move the root file system to a multipath storage device. Therefore you should plan your installation accordingly. See https://access.redhat.com/site/solutions/66501 for more information.

12.17. Advanced Storage Options

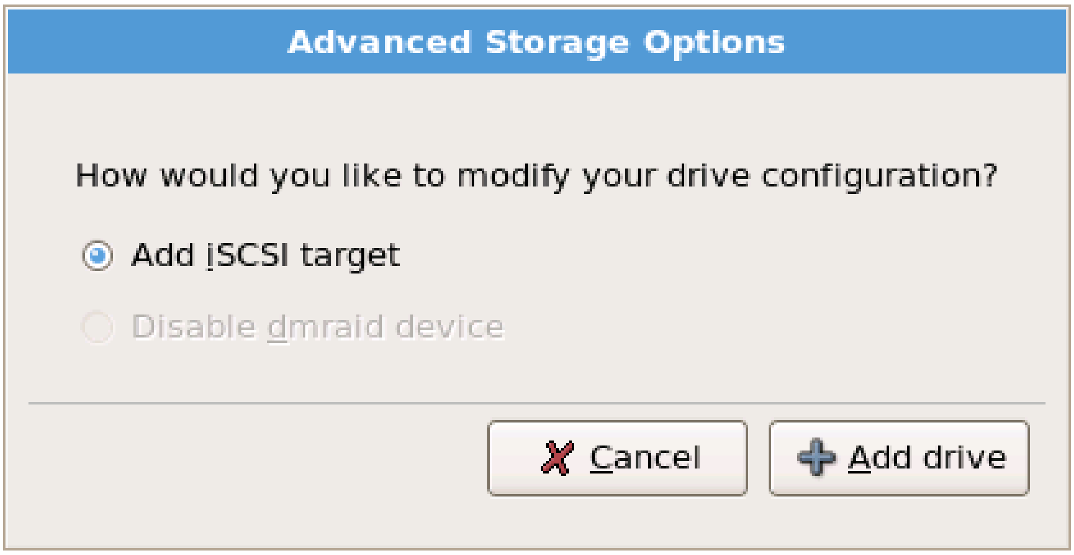

Figure 12.13. Advanced Storage Options

From this screen you can choose to disable a dmraid device, in which case the individual elements of the dmraid device will appear as separate hard drives. You can also choose to configure an iSCSI (SCSI over TCP/IP) target.

To configure an ISCSI target invoke the **`Configure ISCSI Parameters`** dialog by selecting **Add**

`ISCSI target` and clicking on the **`Add Drive`** button. If a network connection is not already active, the installer prompts you to provide details of your network interface. Select your network interface from the drop-down menu, then either leave the **`Use dynamic IP configuration`** box checked, or uncheck it to enter the IP address of your system and the IP addresses of the gateway and nameserver on your network. Ensure that the **`Enable IPv4`** box remains checked.

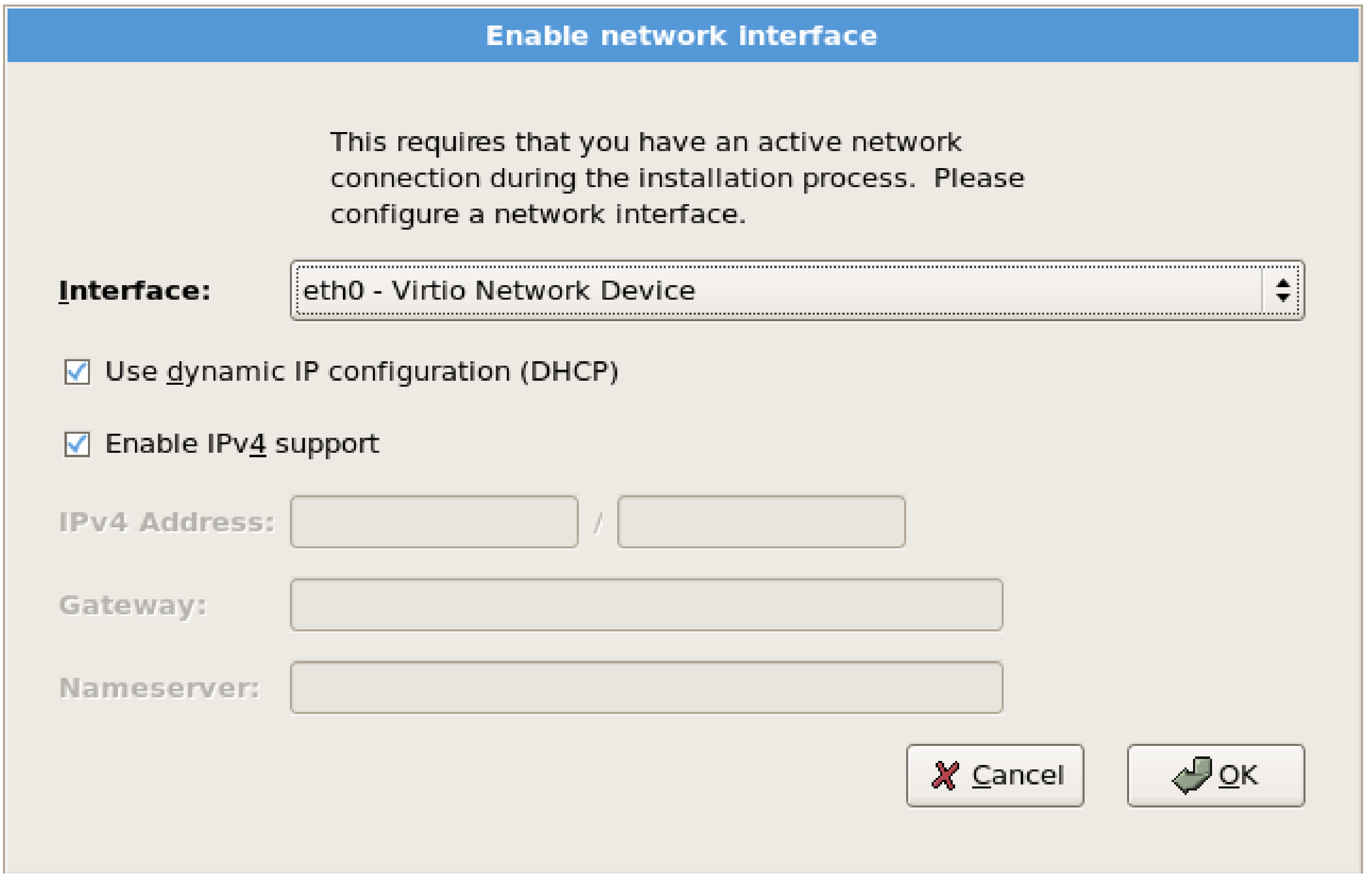

Figure 12.14. Enable network Interface

Fill in the details for the iSCSI target IP and provide a unique iSCSI initiator name to identify this system. If the iSCSI target uses *Challenge Handshake Authentication Protocol* (CHAP) for authentication, enter the CHAP username and password. If your environment uses 2-way CHAP (also called "Mutual CHAP"), also enter the reverse CHAP username and password. Click the **`Add target`** button to attempt connection to the ISCSI target using this information.

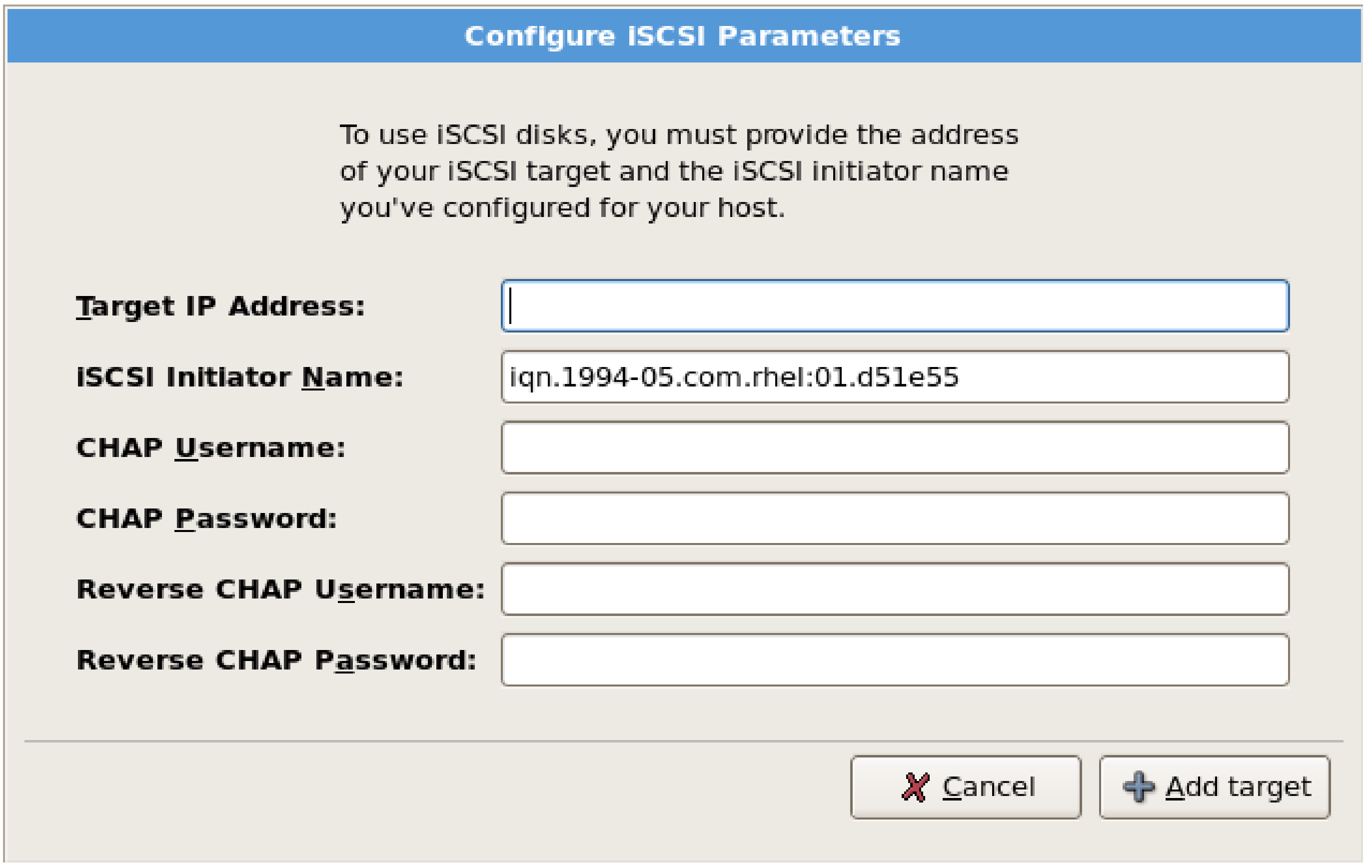

Figure 12.15. Configure ISCSI Parameters

Please note that you will be able to reattempt with a different ISCSI target IP should you enter it incorrectly, but in order to change the ISCSI initiator name you will need to restart the installation.

12.18. Create Default Layout

Create default layout allows you to have some control concerning what data is removed (if any) from your system. Your options are:

- **`Remove all partitions on selected drives and create default layout`** — select this option to remove all partitions on your hard drive(s) (this includes partitions created by other operating systems such as Windows VFAT or NTFS partitions).

> **Warning**
>
> If you select this option, all data on the selected hard drive(s) is removed by the installation program. Do not select this option if you have information that you want to keep on the hard drive(s) where you are installing Red Hat Enterprise Linux.

- **`Remove Linux partitions on selected drives and create default layout`** — select this option to remove only Linux partitions (partitions created from a previous Linux installation). This does not remove other partitions you may have on your hard drive(s) (such as VFAT or FAT32 partitions).

- **`Use free space on selected drives and create default layout`** — select this option to retain your current data and partitions, assuming you have enough free space available on your hard drive(s).

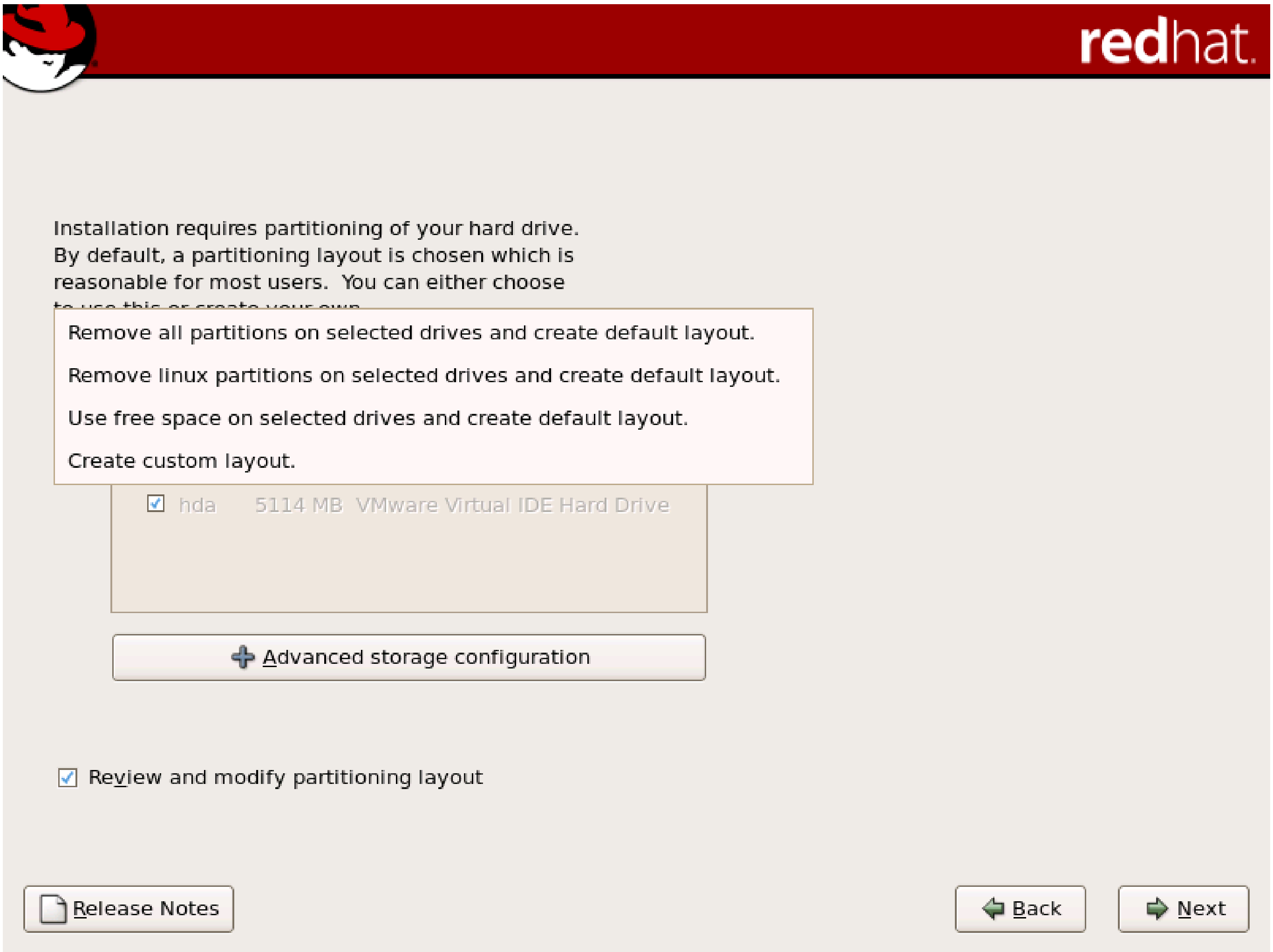

Figure 12.16. Create Default Layout

Using your mouse, choose the storage drive(s) on which you want Red Hat Enterprise Linux to be installed. If you have two or more drives, you can choose which drive(s) should contain this installation. Unselected drives, and any data on them, are not touched.

> **Warning**
>
> It is always a good idea to back up any data that you have on your systems. For example, if you are upgrading or creating a dual-boot system, you should back up any data you wish to keep on your drive(s). Mistakes do happen and can result in the loss of all your data.

> **Note**
>
> [This text does not apply to iSeries system users.]
>
> If you have a RAID card, be aware that some BIOSes do not support booting from the RAID card. In cases such as these, the **`/boot/`** partition must be created on a partition outside of the RAID array, such as on a separate hard drive. An internal hard drive is necessary to use for partition creation with problematic RAID cards.
>
> A **`/boot/`** partition is also necessary for software RAID setups.
>
> If you have chosen to automatically partition your system, you should select **`Review`** and manually edit your **`/boot/`** partition.

To review and make any necessary changes to the partitions created by automatic partitioning, select the **`Review`** option. After selecting **`Review`** and clicking **`Next`** to move forward, the partitions created for you in **Disk Druid** appear. You can make modifications to these partitions if they do not meet your needs.

Click **`Next`** once you have made your selections to proceed.

12.19. Partitioning Your System

If you chose one of the three automatic partitioning options and did not select **`Review`**, skip ahead to Section 12.20, “Network Configuration”.

If you chose one of the automatic partitioning options and selected **`Review`**, you can either accept the current partition settings (click **`Next`**), or modify the setup using **Disk Druid**, the manual partitioning tool.

> **Note**
>
> Please note that in the text mode installation it is not possible to work with LVM (Logical Volumes) beyond viewing the existing setup. LVM can only be set up using the graphical Disk Druid program in a graphical installation.

If you chose to create a custom layout, you must tell the installation program where to install Red Hat Enterprise Linux. This is done by defining mount points for one or more disk partitions in which Red Hat Enterprise Linux is installed.

> **Note**
>
> If you have not yet planned how to set up your partitions, refer to Chapter 26, *An Introduction to Disk Partitions* and Section 12.19.4, “Recommended Partitioning Scheme”. At a bare minimum, you need an appropriately-sized root (`/`) partition, a **`/boot/`** partition, PPC PReP boot partition, and a swap partition equal to twice the amount of RAM you have on the system.

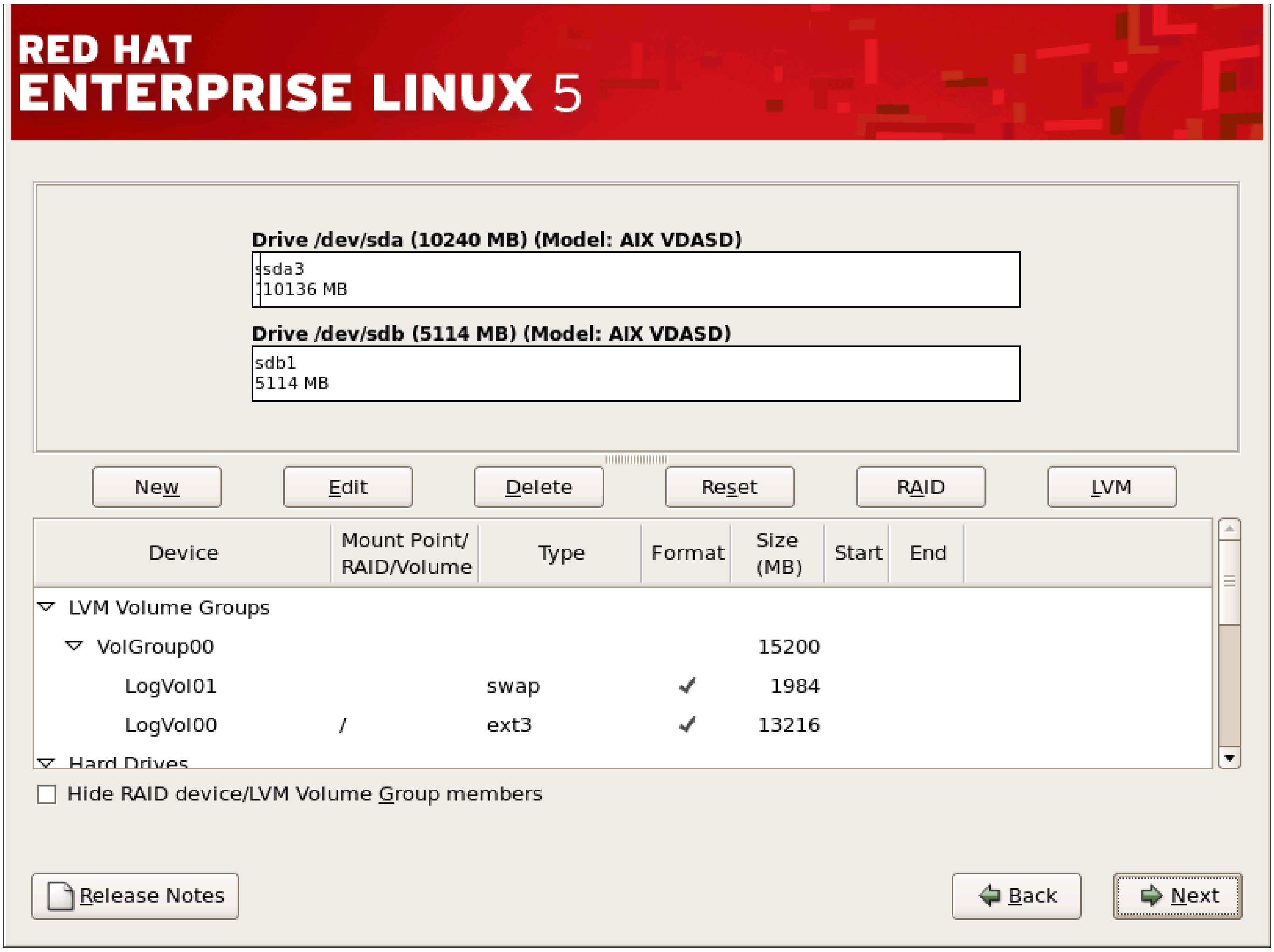

Figure 12.17. Partitioning with Disk Druid on IBM System p and System i systems

The partitioning tool used by the installation program is **Disk Druid**. With the exception of certain esoteric situations, **Disk Druid** can handle the partitioning requirements for a typical installation.

12.19.1. Graphical Display of Hard Drive(s)

Disk Druid offers a graphical representation of your hard drive(s).

Using your mouse, click once to highlight a particular field in the graphical display. Double-click to edit an existing partition or to create a partition out of existing free space.

Above the display, you can review the name of the drive (such as /dev/hda), its size (in MB), and its model as detected by the installation program.

12.19.2. Disk Druid's Buttons

These buttons control **Disk Druid's** actions. They are used to change the attributes of a partition (for example the file system type and mount point) and also to create RAID devices. Buttons on this screen are also used to accept the changes you have made, or to exit **Disk Druid**. For further explanation, take a look at each button in order:

- **New**: Used to request a new partition. When selected, a dialog box appears containing fields (such as the mount point and size fields) that must be filled in.

- **Edit**: Used to modify attributes of the partition currently selected in the **Partitions** section. Selecting **Edit** opens a dialog box. Some or all of the fields can be edited, depending on whether the partition information has already been written to disk.

  You can also edit free space as represented in the graphical display to create a new partition within that space. Either highlight the free space and then select the **Edit** button, or double-click on the free space to edit it.

- To make a RAID device, you must first create (or reuse existing) software RAID partitions. Once you have created two or more software RAID partitions, select **Make RAID** to join the software RAID partitions into a RAID device.

- **Delete**: Used to remove the partition currently highlighted in the **Current Disk Partitions** section. You will be asked to confirm the deletion of any partition.

- **Reset**: Used to restore **Disk Druid** to its original state. All changes made will be lost if you **Reset** the partitions.

- **RAID**: Used to provide redundancy to any or all disk partitions. *It should only be used if you have experience using RAID.* To read more about RAID, see the Red Hat Enterprise Linux Deployment Guide.

  To make a RAID device, you must first create software RAID partitions. Once you have created two or more software RAID partitions, select **RAID** to join the software RAID partitions into a RAID device.

- **LVM**: Allows you to create an LVM logical volume. The role of LVM (Logical Volume Manager) is to present a simple logical view of underlying physical storage space, such as a hard drive(s). LVM manages individual physical disks — or to be more precise, the individual partitions present on them. *It should only be used if you have experience using LVM.* To read more about LVM, see the Red Hat Enterprise Linux Deployment Guide. Note, LVM is only available in the graphical installation program.

  To create an LVM logical volume, you must first create partitions of type physical volume (LVM). Once you have created one or more physical volume (LVM) partitions, select **LVM** to create an LVM logical volume.

12.19.3. Partition Fields

Above the partition hierarchy are labels which present information about the partitions you are creating. The labels are defined as follows:

- **Device**: This field displays the partition's device name.

- **Mount Point/RAID/Volume**: A mount point is the location within the directory hierarchy at which a volume exists; the volume is "mounted" at this location. This field indicates where the partition is mounted. If a partition exists, but is not set, then you need to define its mount point. Double-click on the partition or click the **Edit** button.

- **Type**: This field shows the partition's file system type (for example, ext2, ext3, or vfat).

- **Format**: This field shows if the partition being created will be formatted.

- **Size (MB)**: This field shows the partition's size (in MB).

- **Start**: This field shows the cylinder on your hard drive where the partition begins.

- **End**: This field shows the cylinder on your hard drive where the partition ends.

`Hide RAID device/LVM Volume Group members`: Select this option if you do not want to view any RAID device or LVM Volume Group members that have been created.

12.19.4. Recommended Partitioning Scheme

Unless you have a reason for doing otherwise, we recommend that you create the following partitions:

- A swap partition (at least 256 MB) — swap partitions are used to support virtual memory. In other words, data is written to a swap partition when there is not enough RAM to store the data your system is processing.

  In years past, the recommended amount of swap space increased linearly with the amount of RAM in the system. But because the amount of memory in modern systems has increased into the hundreds of gigabytes, it is now recognized that the amount of swap space that a system needs is a function of the memory workload running on that system. However, given that swap space is usually designated at install time, and that it can be difficult to determine beforehand the memory workload of a system, we recommend determining system swap using the following table.

  Table 12.2. Recommended System Swap Space

Amount of RAM in the System	Recommended Amount of Swap Space
4GB of RAM or less	a minimum of 2GB of swap space
4GB to 16GB of RAM	a minimum of 4GB of swap space
16GB to 64GB of RAM	a minimum of 8GB of swap space
64GB to 256GB of RAM	a minimum of 16GB of swap space
256GB to 512GB of RAM	a minimum of 32GB of swap space

  Note that you can obtain better performance by distributing swap space over multiple storage devices, particularly on systems with fast drives, controllers, and interfaces.

- A PPC PReP boot partition on the first partition of the hard drive — the PPC PReP boot partition contains the YABOOT boot loader (which allows other POWER systems to boot Red Hat Enterprise Linux). Unless you plan to boot from a floppy or network source, you must have a PPC PReP boot partition to boot Red Hat Enterprise Linux.

  For IBM System i and IBM System p users: The PPC PReP boot partition should be between 4-8 MB, not to exceed 10 MB.

- A **`/boot/`** partition (100 MB) — the partition mounted on **`/boot/`** contains the operating system kernel (which allows your system to boot Red Hat Enterprise Linux), along with files used during the bootstrap process. Due to the limitations of most PC firmware, creating a small partition to hold these is a good idea. For most users, a 100 MB boot partition is sufficient.

  Warning

  If you have a RAID card, be aware that Red Hat Enterprise Linux 5.11 does not support setting up hardware RAID on an IPR card. You can boot the standalone diagnostics CD prior to installation to create a RAID array and then install to that RAID array.

- A **`root`** partition (3.0 GB - 5.0 GB) — this is where "**`/`**" (the root directory) is located. In this setup, all files (except those stored in **`/boot`**) are on the root partition.

A 3.0 GB partition allows you to install a minimal installation, while a 5.0 GB root partition lets you perform a full installation, choosing all package groups.

Warning — do not place /var on network storage

Red Hat Enterprise Linux 5.11 does not support having a separate **/var** on a network filesystem (for example, NFS, iSCSI, or NBD) The **/var** directory contains critical data that must be read from or written to during the boot process before establishing network services.

However, you may have **/var/spool**, **/var/www** or other subdirectories on a separate network disk, just not the complete **/var** filesystem.

12.19.5. Adding Partitions

To add a new partition, select the **New** button. A dialog box appears (refer to Figure 12.18, “Creating a New Partition”).

Note

You must dedicate at least one partition for this installation, and optionally more. For more information, refer to Chapter 26, *An Introduction to Disk Partitions*.

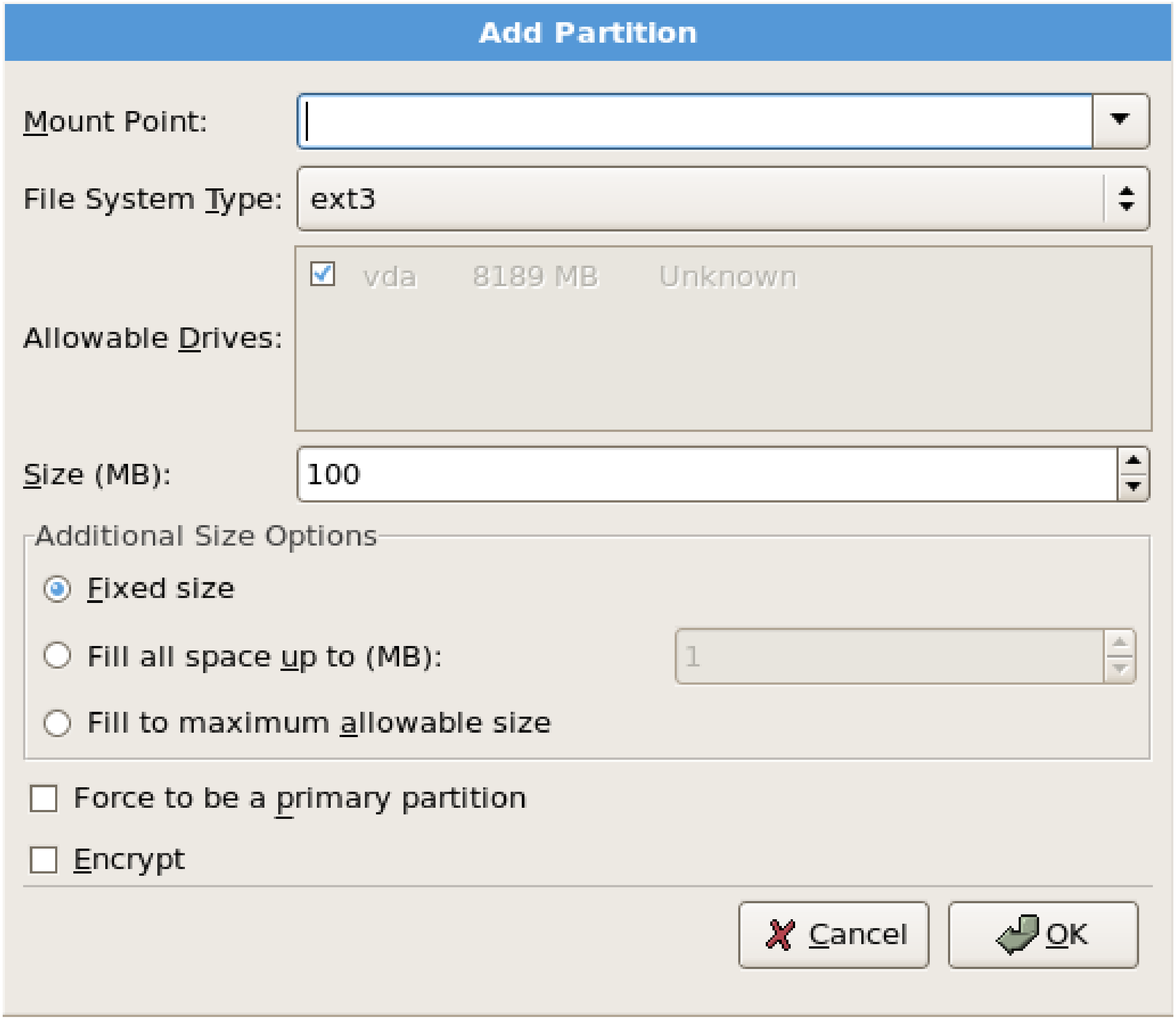

Figure 12.18. Creating a New Partition

- **`Mount Point`**: Enter the partition's mount point. For example, if this partition should be the root partition, enter **`/`**; enter **`/boot`** for the **`/boot`** partition, and so on. You can also use the pull-down menu to choose the correct mount point for your partition. For a swap partition the mount point should not be set - setting the filesystem type to swap is sufficient.

- **`File System Type`**: Using the pull-down menu, select the appropriate file system type for this partition. For more information on file system types, refer to Section 12.19.5.1, “File System Types”.

- **`Allowable Drives`**: This field contains a list of the hard disks installed on your system. If a hard disk's box is highlighted, then a desired partition can be created on that hard disk. If the box is *not* checked, then the partition will *never* be created on that hard disk. By using different checkbox settings, you can have **Disk Druid** place partitions where you need them, or let **Disk Druid** decide where partitions should go.

- **`Size (MB)`**: Enter the size (in megabytes) of the partition. Note, this field starts with 100 MB; unless changed, only a 100 MB partition will be created.

- **`Additional Size Options`**: Choose whether to keep this partition at a fixed size, to allow it to "grow" (fill up the available hard drive space) to a certain point, or to allow it to grow to fill any remaining hard drive space available.

If you choose **Fill all space up to (MB)**, you must give size constraints in the field to the right of this option. This allows you to keep a certain amount of space free on your hard drive for future use.

- **Force to be a primary partition**: Select whether the partition you are creating should be one of the first four partitions on the hard drive. If unselected, the partition is created as a logical partition. Refer to Section 26.1.3, “Partitions within Partitions — An Overview of Extended Partitions”, for more information.
- **Encrypt**: Choose whether to encrypt the partition so that the data stored on it cannot be accessed without a passphrase, even if the storage device is connected to another system. Refer to Chapter 29, *Disk Encryption Guide* for information on encryption of storage devices. If you select this option, the installer prompts you to provide a passphrase before it writes the partition to the disk.
- **OK**: Select **OK** once you are satisfied with the settings and wish to create the partition.
- **Cancel**: Select **Cancel** if you do not want to create the partition.

12.19.5.1. File System Types

Red Hat Enterprise Linux allows you to create different partition types, based on the file system they will use. The following is a brief description of the different file systems available, and how they can be utilized.

- **ext3** — The ext3 file system is based on the ext2 file system and has one main advantage — journaling. Using a journaling file system reduces time spent recovering a file system after a crash as there is no need to **fsck** [6] the file system. A maximum file system size of 16TB is supported for ext3. The ext3 file system is selected by default and is highly recommended.
- **ext2** — An ext2 file system supports standard Unix file types (regular files, directories, symbolic links, etc). It provides the ability to assign long file names, up to 255 characters.
- **physical volume (LVM)** — Creating one or more physical volume (LVM) partitions allows you to create an LVM logical volume. LVM can improve performance when using physical disks. For more information regarding LVM, see the Red Hat Enterprise Linux Deployment Guide.
- **software RAID** — Creating two or more software RAID partitions allows you to create a RAID device. For more information regarding RAID, see chapter *RAID (Redundant Array of Independent Disks)* in the Red Hat Enterprise Linux Deployment Guide.
- **swap** — Swap partitions are used to support virtual memory. In other words, data is written to a swap partition when there is not enough RAM to store the data your system is processing. See the Red Hat Enterprise Linux Deployment Guide for additional information.

12.19.6. Editing Partitions

To edit a partition, select the **Edit** button or double-click on the existing partition.

> **Note**
>
> If the partition already exists on your disk, you can only change the partition's mount point. To make any other changes, you must delete the partition and recreate it.

12.20. Network Configuration

If you do not have a network device, physical LAN, or virtual LAN, this screen does not appear during your installation and you should advance to Section 12.21, “Time Zone Configuration”.

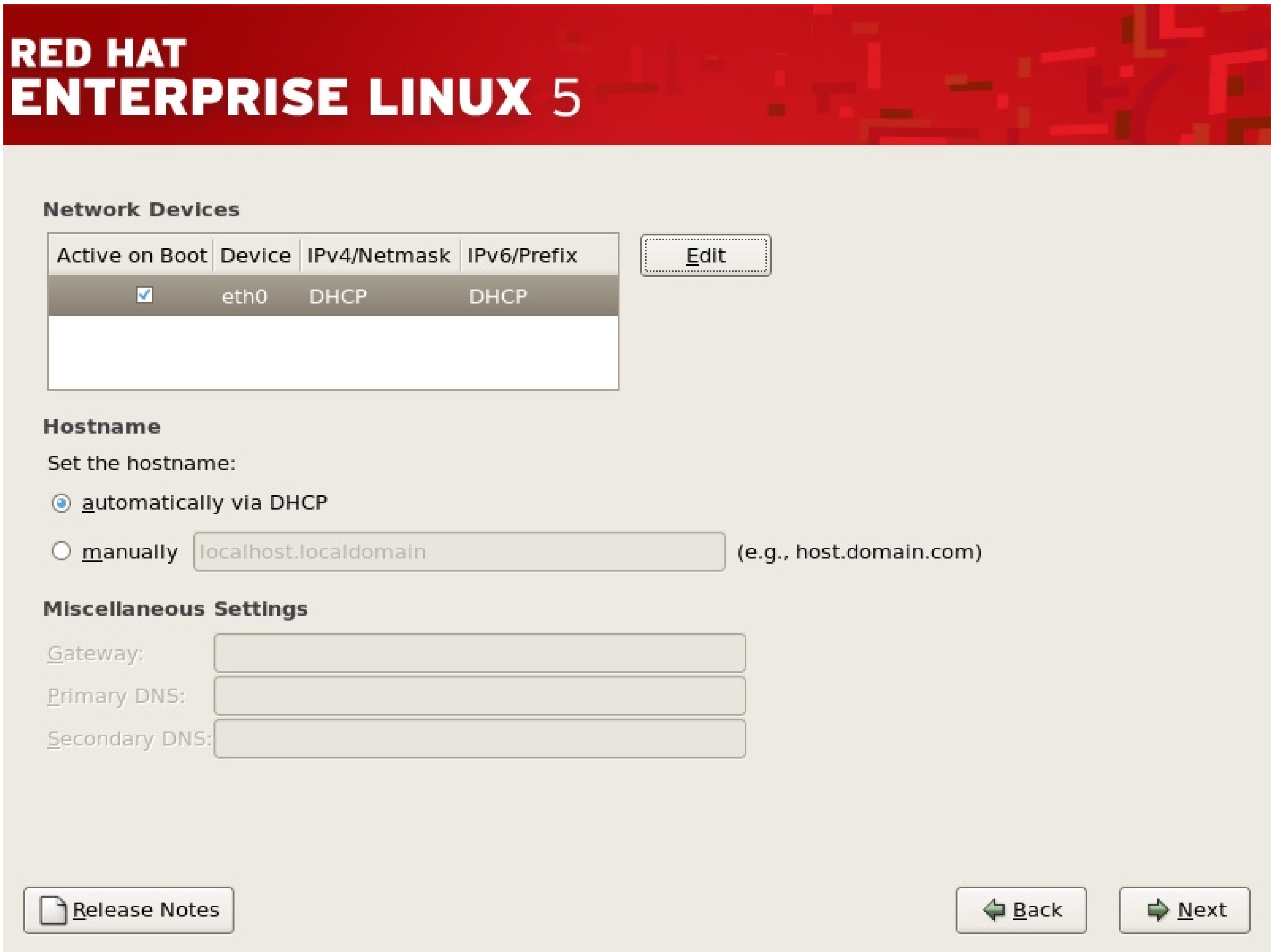

Figure 12.19. Network Configuration

The installation program automatically detects any network devices you have and displays them in the **`Network Devices`** list.

When you have selected a network device, click **`Edit`**. From the **`Edit Interface`** dialog, you can choose to configure the IP address and Netmask (for IPv4 - Prefix for IPv6) of the device to use DHCP or to use static settings. If you do not have DHCP client access or you are unsure what to provide here, contact your network administrator.

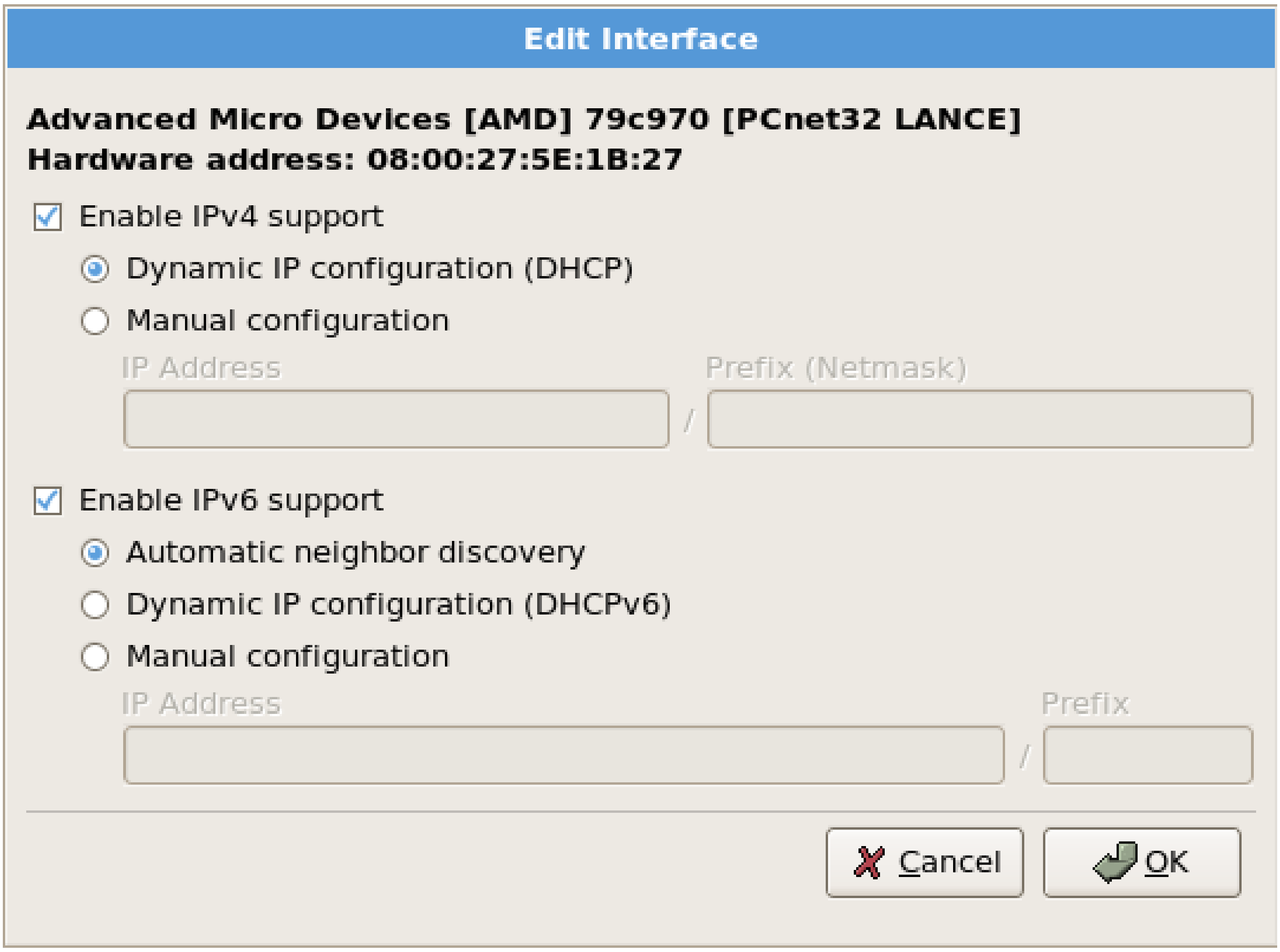

Figure 12.20. Editing a Network Device

Note

Do not use the numbers as seen in this sample configuration. These values will not work for your own network configuration. If you are not sure what values to enter, contact your network administrator for assistance.

If you have a hostname (fully qualified domain name) for the network device, you can choose to have DHCP (Dynamic Host Configuration Protocol) automatically detect it or you can manually enter the hostname in the field provided.

Finally, if you entered the IP and Netmask information manually, you may also enter the Gateway address and the Primary and Secondary DNS addresses.

Note

To change your network configuration after you have completed the installation, use the **Network Administration Tool**.

Type the **`system-config-network`** command in a shell prompt to launch the **Network Administration Tool**. If you are not root, it prompts you for the root password to continue.

12.21. Time Zone Configuration

Set your time zone by selecting the city closest to your computer's physical location. Click on the map to zoom in to a particular geographical region of the world.

From here there are two ways for you to select your time zone:

- Using your mouse, click on the interactive map to select a specific city (represented by a yellow dot). A red **X** appears indicating your selection.
- You can also scroll through the list at the bottom of the screen to select your time zone. Using your mouse, click on a location to highlight your selection.

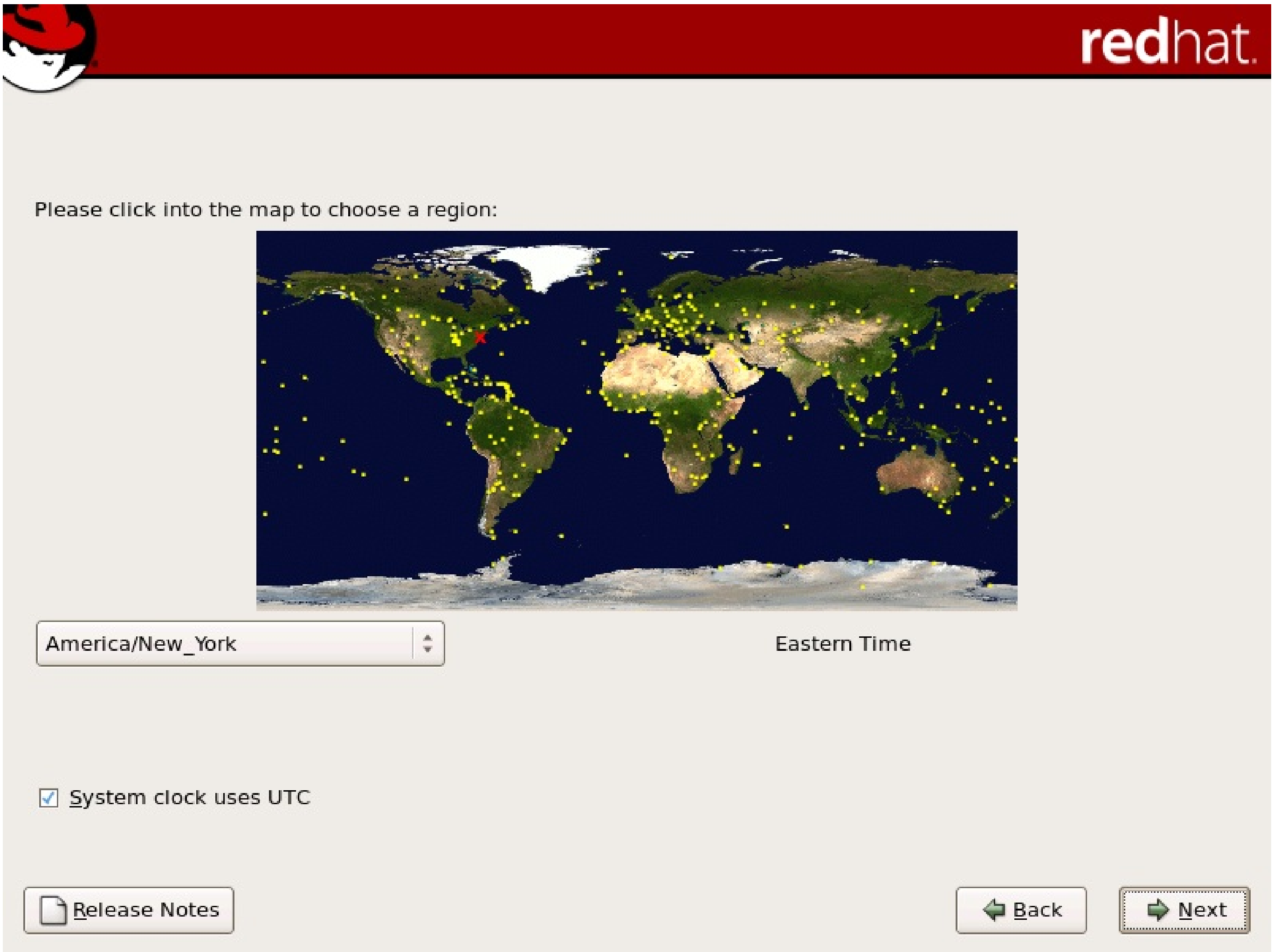

Figure 12.21. Configuring the Time Zone

Select **`System Clock uses UTC`** if you know that your system is set to UTC.

Note

To change your time zone configuration after you have completed the installation, use the **Time and Date Properties Tool**.

Type the **`system-config-date`** command in a shell prompt to launch the **Time and Date Properties Tool**. If you are not root, it prompts you for the root password to continue.

To run the **Time and Date Properties Tool** as a text-based application, use the command **`timeconfig`**.

12.22. Set Root Password

Setting up a root account and password is one of the most important steps during your installation. Your root account is similar to the administrator account used on Windows NT machines. The root account is used to install packages, upgrade RPMs, and perform most system maintenance. Logging in as root gives you complete control over your system.

Note

The root user (also known as the superuser) has complete access to the entire system; for this reason, logging in as the root user is best done *only* to perform system maintenance or administration.

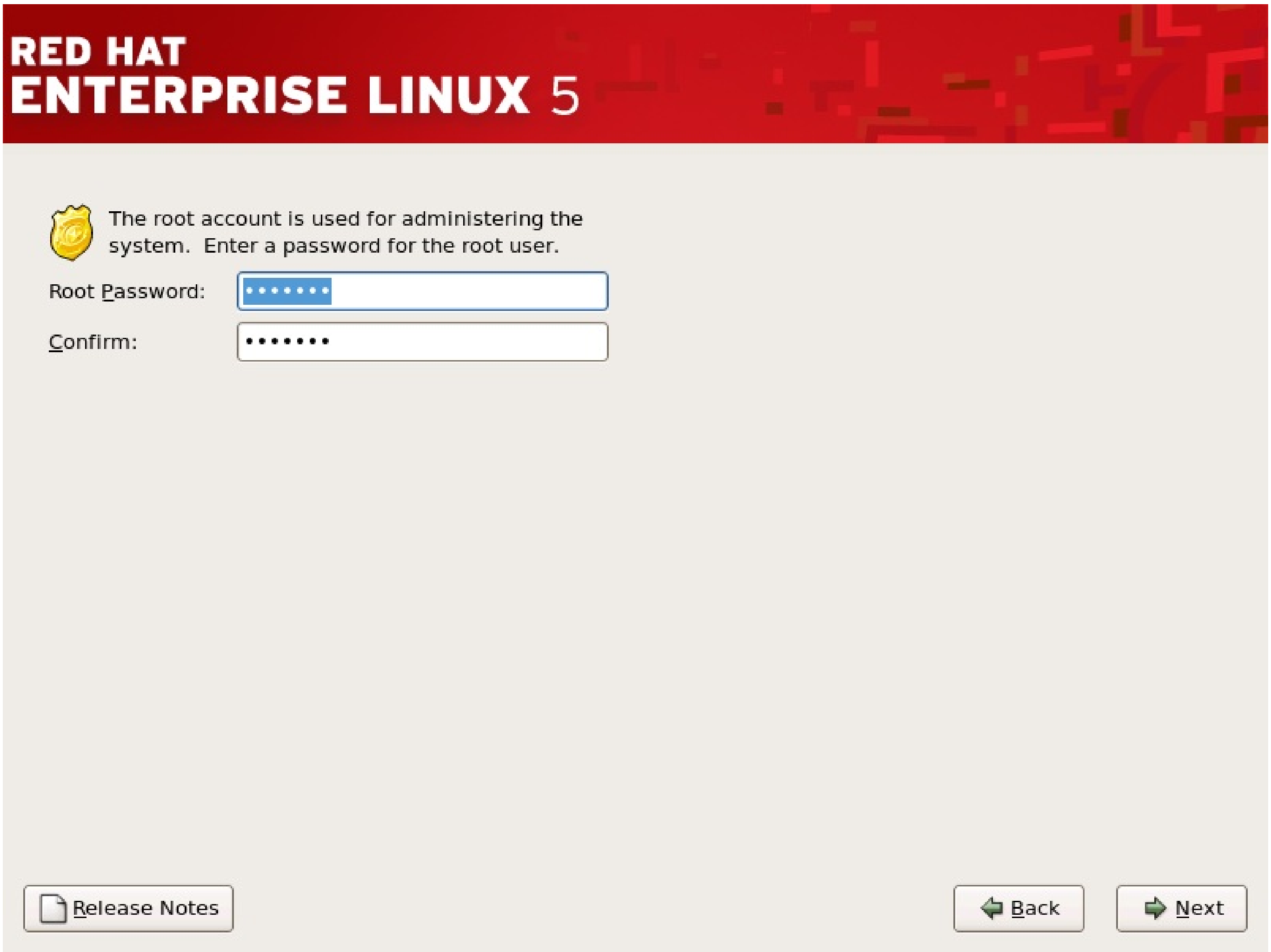

Figure 12.22. Root Password

Use the root account only for system administration. Create a non-root account for your general use and **su -** to root when you need to fix something quickly. These basic rules minimize the chances of a typo or an incorrect command doing damage to your system.

Note

To become root, type **su -** at the shell prompt in a terminal window and then press **Enter**. Then, enter the root password and press **Enter**.

The installation program prompts you to set a root password [7] for your system. *You cannot proceed to the next stage of the installation process without entering a root password.*

The root password must be at least six characters long; the password you type is not echoed to the screen. You must enter the password twice; if the two passwords do not match, the installation program asks you to enter them again.

You should make the root password something you can remember, but not something that is easy for someone else to guess. Your name, your phone number, *qwerty*, *password, root*, *123456*, and *anteater* are all examples of bad passwords. Good passwords mix numerals with upper and lower case letters and do not contain dictionary words: *Aard387vark* or *420BMttNT*, for example. Remember that the password is case-sensitive. If you write down your password, keep it in a secure place. However, it is recommended that you do not write down this or any password you create.

> **Note**
>
> Do not use one of the example passwords offered in this manual. Using one of these passwords could be considered a security risk.

> **Note**
>
> To change your root password after you have completed the installation, use the **Root Password Tool**.
>
> Type the **`system-config-rootpassword`** command in a shell prompt to launch the **Root Password Tool**. If you are not root, it prompts you for the root password to continue.

12.23. Package Group Selection

Now that you have made most of the choices for your installation, you are ready to confirm the default package selection or customize packages for your system.

The **`Package Installation Defaults`** screen appears and details the default package set for your Red Hat Enterprise Linux installation. This screen varies depending on the version of Red Hat Enterprise Linux you are installing.

If you choose to accept the current package list, skip ahead to Section 12.24, “Preparing to Install”.

To customize your package set further, select the **`Customize now`** option on the screen. Clicking **`Next`** takes you to the **`Package Group Selection`** screen.

You can select package groups, which group components together according to function (for example, **`X Window System`** and **`Editors`**), individual packages, or a combination of the two.

> **Note**
>
> Users who want support for developing or running 64-bit applications are encouraged to select the **`Compatibility Arch Support`** and **`Compatibility Arch Development Support`** packages to install architecture specific support for their systems.

To select a component, click on the checkbox beside it (refer to Figure 12.23, “Package Group Selection”).

Figure 12.23. Package Group Selection

Select each component you wish to install.

Once a package group has been selected, if optional components are available you can click on **`Optional packages`** to view which packages are installed by default, and to add or remove optional packages from that group. If there are no optional components this button will be disabled.

Figure 12.24. Package Group Details

12.24. Preparing to Install

12.24.1. Prepare to Install

A screen preparing you for the installation of Red Hat Enterprise Linux now appears.

For your reference, a complete log of your installation can be found in **`/root/install.log`** once you reboot your system.

> **Warning**
>
> If, for some reason, you would rather not continue with the installation process, this is your last opportunity to safely cancel the process and reboot your machine. Once you press the **`Next`** button, partitions are written and packages are installed. If you wish to abort the installation, you should reboot now before any existing information on any hard drive is rewritten.

To cancel this installation process, press your computer's Reset button or use the **`Control`**+**`Alt`**+**`Delete`** key combination to restart your machine.

12.25. Installing Packages

At this point there is nothing left for you to do until all the packages have been installed. How quickly this happens depends on the number of packages you have selected and your computer's speed.

12.26. Installation Complete

Congratulations! Your Red Hat Enterprise Linux installation is now complete!

The installation program prompts you to prepare your system for reboot.

IBM eServer System p and System i — Completing the Installation

Do not forget to remove any boot media.

After rebooting, you must set the open firmware boot device to the disk containing your Red Hat Enterprise Linux PReP and / partitions. To accomplish this, wait until the LED indicator or HMC SRC says **E1F1**, then press **1** to enter the System Management Services GUI. Click on **Select Boot Options**. Select **Select Boot Devices**. Select **Configure 1st Boot Device**. Select the disk containing Red Hat Enterprise Linux. Set the other devices as you wish. Then exit the SMS menus to boot your new system.

> **Note**
>
> Steps in the SMS menu may be different depending on the machine model.

After your computer's normal power-up sequence has completed, YABOOT's prompt appears, at which you can do any of the following things:

- Press **Enter** — causes YABOOT's default boot entry to be booted.
- Select a boot label, followed by **Enter** — causes YABOOT to boot the operating system corresponding to the boot label. (Press **Tab** for non-System i systems at the **boot:** prompt for a list of valid boot labels.)
- Do nothing — after YABOOT's timeout period, (by default, five seconds) YABOOT automatically boots the default boot entry.

Once Red Hat Enterprise Linux has booted, one or more screens of messages should scroll by. Eventually, a **login:** prompt or a GUI login screen (if you installed the X Window System and chose to start X automatically) appears.

The first time you start your Red Hat Enterprise Linux system in run level 5 (the graphical run level), the **Setup Agent** is presented, which guides you through the Red Hat Enterprise Linux configuration. Using this tool, you can set your system time and date, install software, register your machine with Red Hat Network, and more. The **Setup Agent** lets you configure your environment at the beginning, so that you can get started using your Red Hat Enterprise Linux system quickly.

For information on registering your Red Hat Enterprise Linux subscription, refer to Chapter 25, *Register Your System and Apply Subscriptions*.

---

[6] The **fsck** application is used to check the file system for metadata consistency and optionally repair one or more Linux file systems.

[7] A root password is the administrative password for your Red Hat Enterprise Linux system. You should only log in as root when needed for system maintenance. The root account does not operate within the restrictions placed on normal user accounts, so changes made as root can have implications for your entire system.

Chapter 13. Updating drivers during installation on IBM POWER systems

In most cases, Red Hat Enterprise Linux already includes drivers for the devices that make up your system. However, if your system contains hardware that has been released very recently, drivers for this hardware might not yet be included. Sometimes, a driver update that provides support for a new device might be available from Red Hat or your hardware vendor as a *ISO image file* or a *rpm package*. Both these formats supply all the files that make up the driver update together in a single file.

Often, you do not need the new hardware during the installation process. For example, if you use a DVD to install to a local hard drive, the installation will succeed even if drivers for your network card are not available. In situations like this, complete the installation and add support for the piece of hardware afterward — refer to Section 23.1, “Driver update rpm packages” for details of using a driver update rpm package to add this support.

In other situations, you might want to add drivers for a device during the installation process to support a particular configuration. For example, you might want to install drivers for a network device or a storage adapter card to give the installer access to the storage devices that your system uses. You can use a driver update image file to add this support during installation in one of three ways:

1. place the image file in a location accessible to the installer:
    a. on a local IDE hard drive
    b. a USB storage device such as a USB flash drive
    c. on a FTP, HTTP, or NFS server on your local network (or take note of a location on the Internet where someone else has placed the image file)
2. create a *driver update disk* by unpacking the image file onto:
    a. a CD (if your computer has an IDE optical drive)
    b. a DVD (if your computer has an IDE optical drive)
    c. a floppy disk
    d. a USB storage device such as a USB flash drive
3. create an *initial ramdisk update* from the image file and store it on a PXE server. This is an advanced procedure that you should consider only if you cannot perform a driver update with any other method.

If Red Hat, your hardware vendor, or a trusted third party told you that you will require a driver update during the installation process, choose a method to supply the update from the methods described in this chapter and test it before beginning the installation. Conversely, do not perform a driver update during installation unless you are certain that your system requires it. Although installing an unnecessary driver update will not cause harm, the presence of a driver on a system for which it was not intended can complicate support.

13.1. Limitations of driver updates during installation

Unfortunately, some situations persist in which you cannot use a driver update to provide drivers during installation:

Devices already in use

You cannot use a driver update to replace drivers that the installation program has already loaded. Instead, you must complete the installation with the drivers that the installation program loaded and update to the new drivers after installation, or, if you need the new drivers for the installation process, consider performing an initial RAM disk driver update — refer to Section 13.2.3, “Preparing an initial RAM disk update”.

Devices with an equivalent device available

Because all devices of the same type are initialized together, you cannot update drivers for a device if the installation program has loaded drivers for a similar device. For example, consider a system that has two different network adapters, one of which has a driver update available. The installation program will initialize both adapters at the same time, and therefore, you will not be able to use this driver update. Again, complete the installation with the drivers loaded by the installation program and update to the new drivers after installation, or use an initial RAM disk driver update.

13.2. Preparing for a driver update during installation

If a driver update is necessary and available for your hardware, Red Hat or a trusted third party such as the hardware vendor will provide it in the form of an image file in ISO format. Some methods of performing a driver update require you to make the image file available to the installation program, others require you to use the image file to make a driver update disk, and one requires you to prepare an initial RAM disk update:

Methods that use the image file itself

- local hard drive (IDE only)
- USB storage device (for example, USB flash drive)
- network (HTTP, FTP, NFS)

Methods that use a driver update disk produced from an image file

- floppy disk
- CD (IDE only)
- DVD (IDE only)
- USB storage device (for example, USB flash drive)

Methods that use an initial RAM disk update

- PXE

Choose a method to provide the driver update, and refer to Section 13.2.1, “Preparing to use a driver update image file”, Section 13.2.2, “Preparing a driver update disk” or Section 13.2.3, “Preparing an initial RAM disk update”. Note that you can use a USB storage device either to provide an image file, or as a driver update disk.

SATA and SCSI

During installation, you cannot provide driver updates on media that is read by a device with a SATA or SCSI connection. For example, if the only optical drive on your system is a DVD drive with a SATA connection, you cannot provide a driver update on CD or DVD.

13.2.1. Preparing to use a driver update image file

13.2.1.1. Preparing to use an image file on local storage

To make the ISO image file available on local storage, such as a USB flash drive, USB hard drive, or local IDE hard drive, simply copy the file onto the storage device. You can rename the file if you find it helpful to do so, but you must not change the filename extension, which must remain **`.iso`**. In the following example, the file is named **`dd.iso`**:

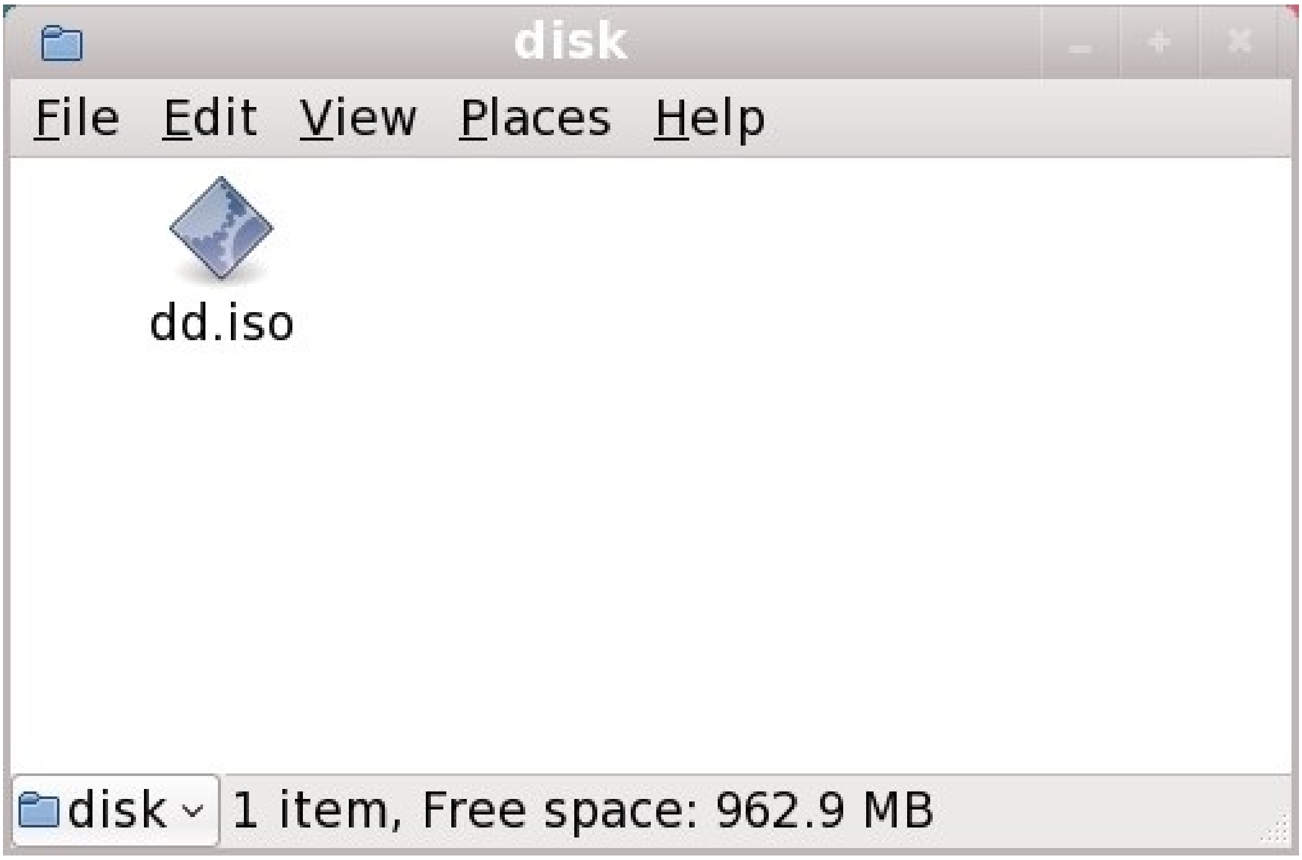

Figure 13.1. Content of a USB flash drive holding a driver update image file

Note that if you use this method, the storage device will contain only a single file. This differs from driver update disks on formats such as CD and DVD, which contain many files. The ISO image file contains all of the files that would normally be on a driver update disk.

Refer to Section 13.3.2, “Let the installer prompt you for a driver update” and Section 13.3.3, “Use a boot option to specify a driver update disk” to learn how to use the driver update disk during installation.

If you change the file system label of the device to **`OEMDRV`**, the installation program will automatically examine it for driver updates and load any that it detects. This behavior is controlled by the **`dlabel=on`** boot option, which is enabled by default. Refer to Section 13.3.1, “Let the installer automatically find a driver update disk”.

13.2.1.2. Preparing to use an image file available through a network

To make the ISO image file available on a local network, place it in a publicly accessible folder on a HTTP, FTP, or NFS server. If you plan to use an image file that is already publicly available through the Internet, no special preparation is necessary. In either case, take note of the URL and verify that you can access the file from another machine on your network before commencing installation.

Refer to Section 13.3.2, “Let the installer prompt you for a driver update” and Section 13.3.4, “Use a boot option to specify a driver update image file on a network” to learn how to specify this network location during installation.

13.2.2. Preparing a driver update disk

You can use a variety of media to create a driver update disk, including CD, DVD, floppy disk, and USB storage devices such as USB flash drives

13.2.2.1. Creating a driver update disk on CD or DVD

These instructions assume that you use the GNOME desktop

CD/DVD Creator is part of the GNOME desktop. If you use a different Linux desktop, or a different operating system altogether, you will need to use another piece of software to create the CD or DVD. The steps will be generally similar.

Make sure that the software that you choose can create CDs or DVDs from image files. While this is true of most CD and DVD burning software, exceptions exist. Look for a button or menu entry labeled **`burn from image`** or similar. If your software lacks this feature, or you do not select it, the resulting disk will hold only the image file itself, instead of the contents of the image file.

1. Use the desktop file manager to locate the driver update ISO image file supplied to you by Red Hat or your hardware vendor.

Figure 13.2. A typical .iso file displayed in a file manager window

2. Right-click on this file and choose **`Write to disc`**. You will see a window similar to the following:

Figure 13.3. CD/DVD Creator's Write to Disc dialog

3. Click the **`Write`** button. If a blank disc is not already in the drive, **CD/DVD Creator** will prompt you to insert one.

After you burn a driver update disk CD or DVD, verify that the disk was created successfully by inserting it into your system and browsing to it using the file manager. You should see a list of files similar to the following:

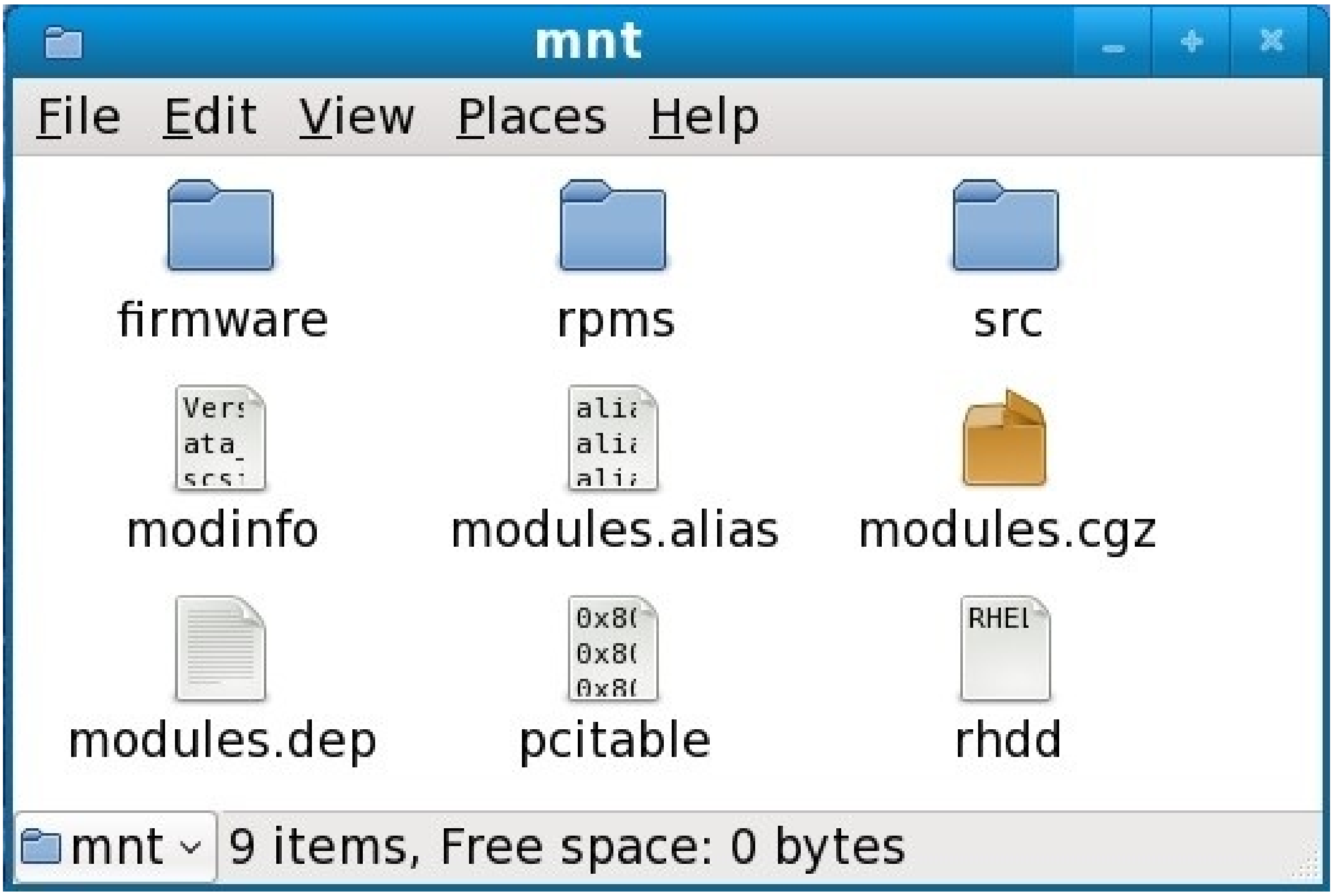

Figure 13.4. Contents of a typical driver update disc on CD or DVD

If you see only a single file ending in **`.iso`**, then you have not created the disk correctly and should try again. Ensure that you choose an option similar to **`burn from image`** if you use a Linux desktop other than GNOME or if you use a different operating system.

Refer to Section 13.3.2, "Let the installer prompt you for a driver update" and Section 13.3.3, "Use a boot option to specify a driver update disk" to learn how to use the driver update disk during installation.

13.2.2.2. Creating a driver update disk on floppy disk, or USB storage device

These instructions assume that you use Linux

The following procedure assumes that you are creating a driver update disk using Linux. To create a driver update disk using a different operating system, you will need to find a tool that can extract files from ISO images. You must then place the extracted files on a removable disk or USB storage device.

These instructions could destroy data

If you perform this procedure with a disk or USB storage device that already contains data, this data will be destroyed with no warning. Make sure that you specify the correct disk or USB storage device, and make sure that this disk or storage device does not contain any data that you want to keep.

1. Insert a blank, formatted floppy disk into an available drive, or connect an empty USB storage device (such as a USB flash drive) to your computer. Note the device name allocated to this disk, for example, **/dev/fd0** for a floppy disk in the first floppy drive on your system.

   If you do not know the device name, become root and use the command **fdisk -l** on the command line. You will see a list of all storage devices available on your system. Compare the output of **fdisk -l** when the disk inserted or the storage device is attached with the output of this command when the disk is removed or the storage device is disconnected.

2. At the command line, change into the directory that contains the image file.

3. At the command line, type:

   ```
   dd if=image of=device
   ```

   where *image* is the image file, and *device* is the device name. For example, to create a driver disk on floppy disk **/dev/fd0** from driver update image file **dd.iso**, you would use:

   ```
   dd if=dd.iso of=/dev/fd0
   ```

After you create a driver update disk, insert it (if you used a disk) or attach it (if you used a USB storage device) and browse to it using the file manager. If you see a list of files similar to those illustrated in Figure 13.4, “Contents of a typical driver update disc on CD or DVD”, you know that you have created the driver update disk correctly.

Refer to Section 13.3.2, “Let the installer prompt you for a driver update” and Section 13.3.3, “Use a boot option to specify a driver update disk” to learn how to use the driver update disk during installation.

13.2.3. Preparing an initial RAM disk update

Advanced procedure

This is an advanced procedure that you should consider only if you cannot perform a driver update with any other method.

The Red Hat Enterprise Linux installation program can load updates for itself early in the installation process from a *RAM disk* — an area of your computer's memory that temporarily behaves as if it were a disk. You can use this same capability to load driver updates. To perform a driver update during installation, your computer must be able to boot from a *preboot execution environment* (PXE) server, and you must have a PXE server available on your network. Refer to Chapter 34, *PXE Network Installations* for instructions on using PXE during installation.

To make the driver update available on your PXE server:

1. Place the driver update image file on your PXE server. Usually, you would do this by downloading it to the PXE server from a location on the Internet specified by Red Hat or your hardware vendor. Names of driver update image files end in **.iso**.

2. Copy the driver update image file into the **/tmp/initrd_update** directory.

3. Rename the driver update image file to **dd.img**.

4. At the command line, change into the **/tmp/initrd_update** directory, type the following command, and press **Enter**:

```
find . | cpio --quiet -c -o | gzip -9 >/tmp/initrd_update.img
```

5. Copy the file **/tmp/initrd_update.img** into the directory the holds the target that you want to use for installation. This directory is placed under the **/tftpboot/pxelinux/** directory. For example, **/tftpboot/pxelinux/r5su3/** might hold the PXE target for Red Hat Enterprise Linux 5.3 Server.

6. Edit the **/tftpboot/pxelinux/pxelinux.cfg/default** file to include an entry that includes the initial RAM disk update that you just created, in the following format:

```
label target-dd
kernel target/vmlinuz
append initrd=target/initrd.img,target/dd.img
```

Where *target* is the target that you want to use for installation.

Example 13.1. Preparing an initial RAM disk update from a driver update image file

In this example, **driver_update.iso** is a driver update image file that you downloaded from the Internet to a directory on your PXE server. The target that you want to PXE boot from is located in **/tftpboot/pxelinux/r5su3**

At the command line, change to the directory that holds the file and enter the following commands:

```
$ cp driver_update.iso /tmp/initrd_update/dd.img
$ cd /tmp/initrd_update
$ find . | cpio --quiet -c -o | gzip -9 >/tmp/initrd_update.img
$ cp /tmp/initrd_update.img /tftpboot/pxelinux/r5su3/dd.img
```

Edit the **/tftpboot/pxelinux/pxelinux.cfg/default** file and include the following entry:

```
label r5su3-dd
kernel r5su3/vmlinuz
append initrd=r5su3/initrd.img,r5su3/dd.img
```

Refer to Section 13.3.5, “Select a PXE target that includes a driver update” to learn how to use an initial RAM disk update during installation.

13.3. Performing a driver update during installation

You can perform a driver update during installation in the following ways:

- let the installer automatically find a driver update disk.
- let the installer prompt you for a driver update.
- use a boot option to specify a driver update disk.
- use a boot option to specify a driver update image file on a network.
- select a PXE target that includes a driver update.

13.3.1. Let the installer automatically find a driver update disk

Attach a block device with the filesystem label **OEMDRV** before starting the installation process. The

installer will automatically examine the device and load any driver updates that it detects and will not prompt you during the process. Refer to Section 13.2.1.1, “Preparing to use an image file on local storage” to prepare a storage device for the installer to find.

13.3.2. Let the installer prompt you for a driver update

1. Begin the installation normally for whatever method you have chosen. If the installer cannot load drivers for a piece of hardware that is essential for the installation process (for example, if it cannot detect any network or storage controllers), it prompts you to insert a driver update disk:

Figure 13.5. The no driver found dialog

2. Select **`Use a driver disk`** and refer to Section 13.4, “Specifying the location of a driver update image file or driver update disk”.

13.3.3. Use a boot option to specify a driver update disk

> **Choose this method only for completely new drivers**
>
> This method only works to introduce completely new drivers, not to update existing drivers.

1. Type **`linux dd`** at the boot prompt at the start of the installation process and press **`Enter`**. The installer prompts you to confirm that you have a driver disk:

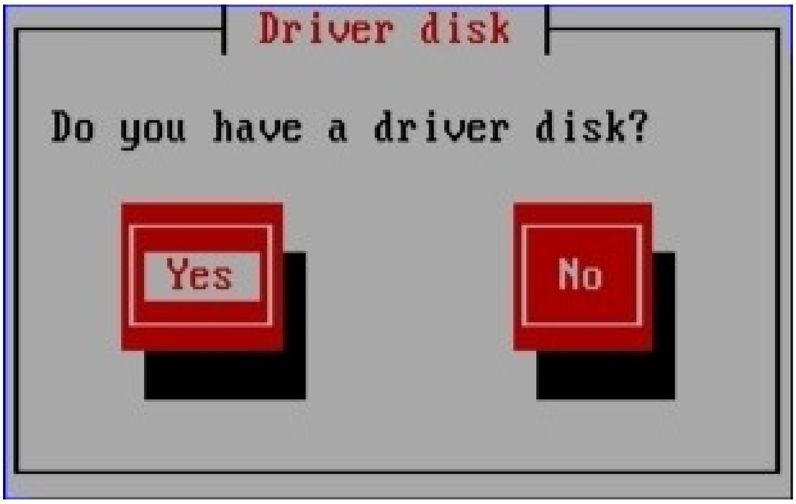

Figure 13.6. The driver disk prompt

2. Insert the driver update disk that you created on CD, DVD, floppy disk, or USB storage device and select **Yes**. The installer examines the storage devices that it can detect. If there is only one possible location that could hold a driver disk (for example, the installer detects the presence of a floppy disk, but no other storage devices) it will automatically load any driver updates that it finds at this location.

   If the installer finds more than one location that could hold a driver update, it prompts you to specify the location of the update. Refer to to Section 13.4, “Specifying the location of a driver update image file or driver update disk” .

13.3.4. Use a boot option to specify a driver update image file on a network

Choose this method only for completely new drivers

This method only works to introduce completely new drivers, not to update existing drivers.

Type **linux dd=*URL*** (where *URL* is the HTTP, FTP, or NFS address of a driver update image) at the boot prompt at the start of the installation process and press **Enter**. The installer will retrieve the driver update image from that address and use it during installation.

13.3.5. Select a PXE target that includes a driver update

1. Select **network boot** in your computer's BIOS or boot menu. The procedure to specify this option varies widely among different computers. Consult your hardware documentation or the hardware vendor for specifics relevant to your computer.

2. In the preexecution boot environment (PXE), choose the boot target that you prepared on your PXE server. For example, if you labeled this environment **r5su3-dd** in the **/tftpboot/pxelinux/pxelinux.cfg/default** file on your PXE server, type **r5su3-dd** at the prompt and press **Enter**.

Refer to Section 13.2.3, “Preparing an initial RAM disk update” and Chapter 34, *PXE Network Installations* for instructions on using PXE to perform an update during installation. Note that this is an advanced procedure — do not attempt it unless other methods of performing a driver update fail.

13.4. Specifying the location of a driver update image file or driver update disk

If the installer detects more than one possible device that could hold a driver update, it prompts you to select the correct device. If you are not sure which option represents the device on which the driver update is stored, try the various options in order until you find the correct one.

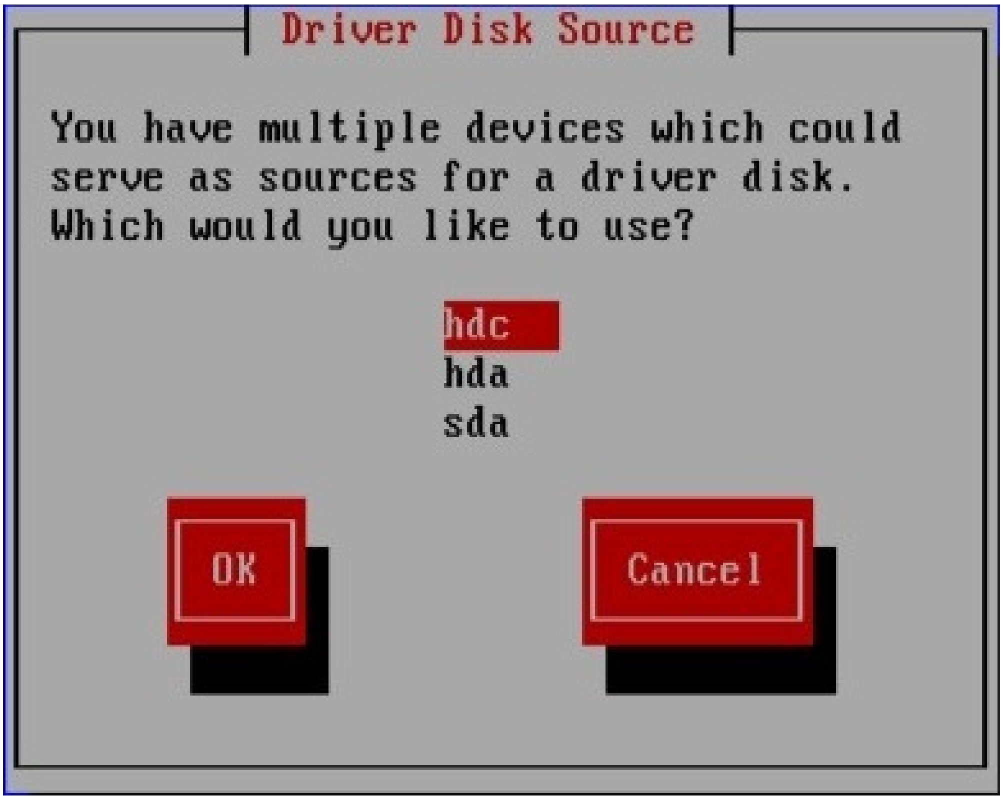

Figure 13.7. Selecting a driver disk source

If the device that you choose contains no suitable update media, the installer will prompt you to make another choice.

If you made a driver update disk on floppy disk, CD, DVD, or USB storage device, the installer now loads the driver update. However, if the device that you selected is a type of device that could contain more than one partition (whether the device currently has more than one partition or not), the installer might prompt you to select the partition that holds the driver update.

Figure 13.8. Selecting a driver disk partition

The installer prompts you to specify which file contains the driver update:

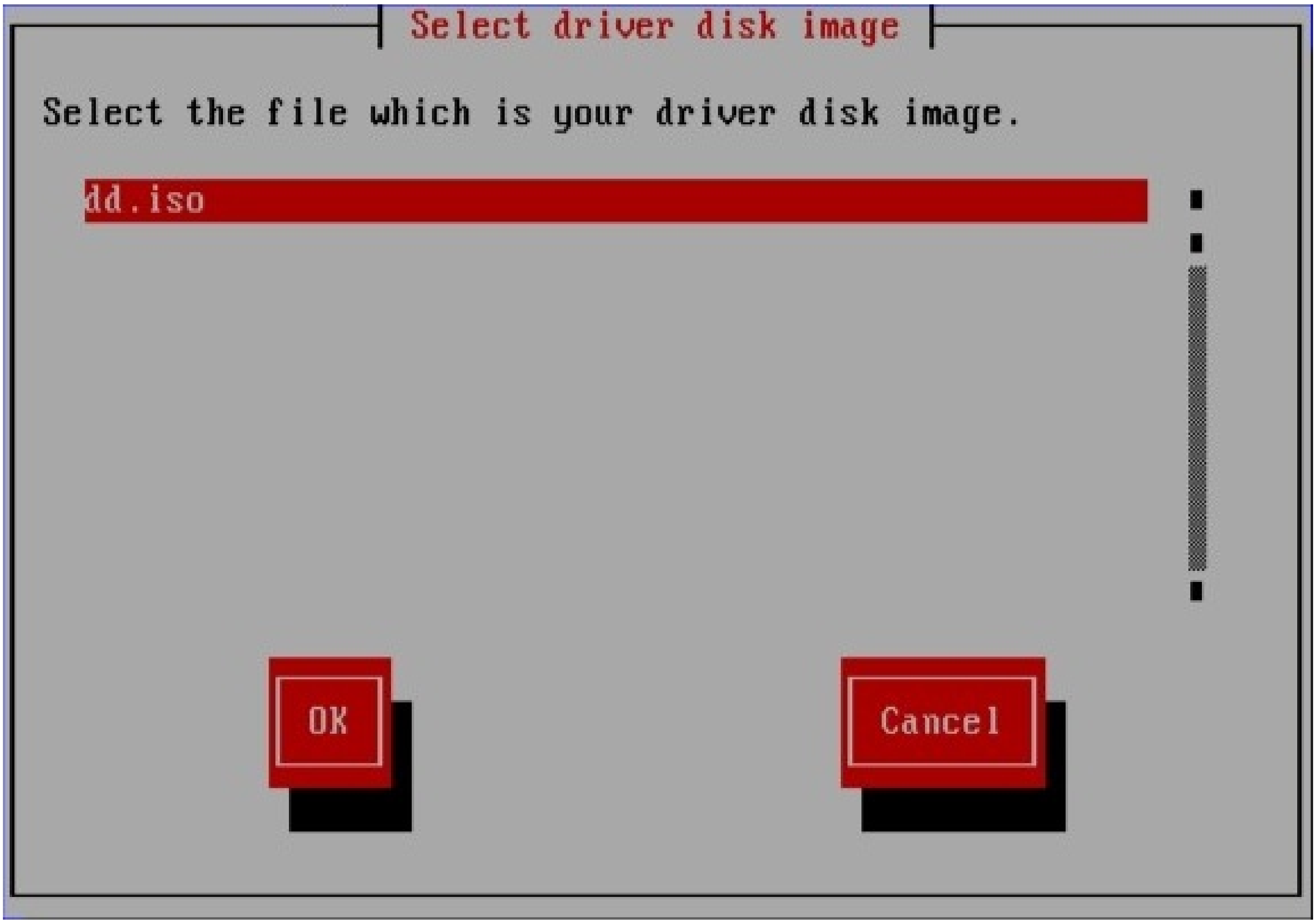

Figure 13.9. Selecting an ISO image

Expect to see these screens if you stored the driver update on an internal hard drive or on a USB storage device. You should not see them if the driver update is on a floppy disk, CD, or DVD.

Regardless of whether you are providing a driver update in the form of an image file or with a driver update disk, the installer now copies the appropriate update files into a temporary storage area (located in system RAM and not on disk). The installer might ask whether you would like to use additional driver updates. If you select **Yes**, you can load additional updates in turn. When you have no further driver updates to load, select **No**. If you stored the driver update on removeable media, you can now safely eject or disconnect the disk or device. The installer no longer requires the driver update, and you can re-use the media for other purposes.

Chapter 14. Troubleshooting Installation on an IBM POWER System

This appendix discusses some common installation problems and their solutions.

You may also find the IBM Online Alert Section for System p and System i useful. It is located at:

```
http://www14.software.ibm.com/webapp/set2/sas/f/lopdiags/info/LinuxAlerts.html
```

Please note that the url above has been split across two lines for readability. It should be entered into a browser as one continuous line, with no linebreak.

14.1. You are Unable to Boot Red Hat Enterprise Linux

14.1.1. Is Your System Displaying Signal 11 Errors?

A signal 11 error, commonly know as a *segmentation fault*, means that the program accessed a memory location that was not assigned to it. A signal 11 error may be due to a bug in one of the software programs that is installed, or faulty hardware.

If you receive a fatal signal 11 error during your installation, it is probably due to a hardware error in memory on your system's bus. Like other operating systems, Red Hat Enterprise Linux places its own demands on your system's hardware. Some of this hardware may not be able to meet those demands, even if they work properly under another OS.

Ensure that you have the latest installation updates and images from Red Hat. Review the online errata to see if newer versions are available. If the latest images still fail, it may be due to a problem with your hardware. Commonly, these errors are in your memory or CPU-cache. A possible solution for this error is turning off the CPU-cache in the BIOS, if your system supports this. You could also try to swap your memory around in the motherboard slots to check if the problem is either slot or memory related.

Another option is to perform a media check on your installation CD-ROMs. The Red Hat Enterprise Linux installation program has the ability to test the integrity of the installation media. It works with the CD, DVD, hard drive ISO, and NFS ISO installation methods. Red Hat recommends that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned CDs). To use this test, type the following command at the **`boot:`** or **`yaboot:`** prompt (prepend with **`elilo`** for Itanium systems):

```
linux mediacheck
```

For more information concerning signal 11 errors, refer to:

```
http://www.bitwizard.nl/sig11/
```

14.2. Trouble Beginning the Installation

14.2.1. Problems with Booting into the Graphical Installation

There are some video cards that have trouble booting into the graphical installation program. If the installation program does not run using its default settings, it tries to run in a lower resolution mode. If that still fails, the installation program attempts to run in text mode.

One possible solution is to try using the **resolution=** boot option. Refer to Chapter 15, *Additional Boot Options for IBM Power Systems* for more information.

> **Note**
>
> To disable frame buffer support and allow the installation program to run in text mode, try using the **nofb** boot option. This command may be necessary for accessibility with some screen reading hardware.

14.3. Trouble During the Installation

14.3.1. No devices found to install Red Hat Enterprise Linux Error Message

If you receive an error message stating **No devices found to install Red Hat Enterprise Linux**, there is probably a SCSI controller that is not being recognized by the installation program.

Check your hardware vendor's website to determine if a driver diskette image is available that fixes your problem. For more general information on driver diskettes, refer to Chapter 13, *Updating drivers during installation on IBM POWER systems*.

You can also refer to the *Red Hat Hardware Compatibility List*, available online at:

http://hardware.redhat.com/hcl/

14.3.2. Saving Traceback Messages Without a Diskette Drive

If you receive a traceback error message during installation, you can usually save it to a diskette.

If you do not have a diskette drive available in your system, you can **scp** the error message to a remote system.

This information does not apply to users of headless IBM System p systems.

When the traceback dialog appears, the traceback error message is automatically written to a file named **/tmp/anacdump.txt**. Once the dialog appears, switch over to a new tty (virtual console) by pressing the keys **Ctrl+Alt+F2** and **scp** the message written to **/tmp/anacdump.txt** to a known working remote system.

14.3.3. Trouble with Partition Tables

If you receive an error after the **Disk Partitioning Setup** (Section 12.16, “Disk Partitioning Setup”) phase of the installation saying something similar to

> The partition table on device hda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive.

you may not have a partition table on that drive or the partition table on the drive may not be recognizable by the partitioning software used in the installation program.

No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

14.3.4. Other Partitioning Problems for IBM™ POWER System Users

If you are using **Disk Druid** to create partitions, but cannot move to the next screen, you probably have not created all the partitions necessary for **Disk Druid**'s dependencies to be satisfied.

You must have the following partitions as a bare minimum:

- A **/** (root) partition
- A <swap> partition of type swap
- A PPC PReP Boot partition.
- A /boot/ partition.

Note

When defining a partition's type as swap, do not assign it a mount point. **Disk Druid** automatically assigns the mount point for you.

14.3.5. Are You Seeing Python Errors?

During some upgrades or installations of Red Hat Enterprise Linux, the installation program (also known as **anaconda**) may fail with a Python or traceback error. This error may occur after the selection of individual packages or while trying to save the upgrade log in the **/tmp/**directory. The error may look similar to:

```
Traceback (innermost last):
File "/var/tmp/anaconda-7.1//usr/lib/anaconda/iw/progress_gui.py", line 20, in run
rc = self.todo.doInstall ()
File "/var/tmp/anaconda-7.1//usr/lib/anaconda/todo.py", line 1468, in doInstall
self.fstab.savePartitions ()
File "fstab.py", line 221, in savePartitions
sys.exit(0)
SystemExit: 0
Local variables in innermost frame:
self: <fstab.GuiFstab instance at 8446fe0>
sys: <module 'sys' (built-in)>
ToDo object:  (itodo  ToDo  p1  (dp2  S'method'  p3  (iimage  CdromInstallMethod
p4  (dp5  S'progressWindow'  p6   <failed>
```

This error occurs in some systems where links to **/tmp/** are symbolic to other locations or have been changed since creation. These symbolic or changed links are invalid during the installation process, so the installation program cannot write information and fails.

If you experience such an error, first try to download any available errata for **anaconda**. Errata can be found at:

```
http://www.redhat.com/support/errata/
```

The **anaconda** website may also be a useful reference and can be found online at:

```
http://fedoraproject.org/wiki/Anaconda
```

You can also search for bug reports related to this problem. To search Red Hat's bug tracking system, go to:

```
http://bugzilla.redhat.com/bugzilla/
```

Finally, if you are still facing problems related to this error, register your product and contact our support team. To register your product, go to:

```
http://www.redhat.com/apps/activate/
```

14.4. Problems After Installation

14.4.1. Unable to IPL from *NWSSTG

If you are experiencing difficulties when trying to IPL from *NWSSTG, you may not have created a PReP Boot partition set as active.

14.4.2. Booting into a Graphical Environment

If you have installed the X Window System but are not seeing a graphical desktop environment once you log into your Red Hat Enterprise Linux system, you can start the X Window System graphical interface using the command **`startx`**.

Once you enter this command and press **Enter**, the graphical desktop environment is displayed.

Note, however, that this is just a one-time fix and does not change the log in process for future log ins.

To set up your system so that you can log in at a graphical login screen, you must edit one file, **`/etc/inittab`**, by changing just one number in the runlevel section. When you are finished, reboot the computer. The next time you log in, you are presented with a graphical login prompt.

Open a shell prompt. If you are in your user account, become root by typing the **`su`** command.

Now, type **`gedit /etc/inittab`** to edit the file with **`gedit`**. The file **`/etc/inittab`** opens. Within the first screen, a section of the file which looks like the following appears:

```
# Default runlevel. The runlevels used by RHS are:
#   0 - halt (Do NOT set initdefault to this)
#   1 - Single user mode
#   2 - Multiuser, without NFS (The same as 3, if you do not have networking)
#   3 - Full multiuser mode
#   4 - unused
#   5 - X11
#   6 - reboot (Do NOT set initdefault to this)
#  id:3:initdefault:
```

To change from a console to a graphical login, you should change the number in the line **`id:3:initdefault:`** from a **3** to a **5**.

Warning

Change *only* the number of the default runlevel from **3** to **5**.

Your changed line should look like the following:

```
id:5:initdefault:
```

When you are satisfied with your change, save and exit the file using the **Ctrl**+**Q** keys. A window appears and asks if you would like to save the changes. Click **Save**.

The next time you log in after rebooting your system, you are presented with a graphical login prompt.

14.4.3. Problems with the X Window System (GUI)

If you are having trouble getting X (the X Window System) to start, you may not have installed it during your installation.

If you want X, you can either install the packages from the Red Hat Enterprise Linux CD-ROMs or perform an upgrade.

If you elect to upgrade, select the X Window System packages, and choose GNOME, KDE, or both, during the upgrade package selection process.

14.4.4. Problems with the X Server Crashing and Non-Root Users

If you are having trouble with the X server crashing when anyone other than root logs in, you may have a full file system (or, a lack of available hard drive space).

To verify that this is the problem you are experiencing, run the following command:

```
df -h
```

The **df** command should help you diagnose which partition is full. For additional information about **df** and an explanation of the options available (such as the **-h** option used in this example), refer to the **df** man page by typing **man df** at a shell prompt.

A key indicator is 100% full or a percentage above 90% or 95% on a partition. The **/home/** and **/tmp/** partitions can sometimes fill up quickly with user files. You can make some room on that partition by removing old files. After you free up some disk space, try running X as the user that was unsuccessful before.

14.4.5. Problems When You Try to Log In

If you did not create a user account in the **Setup Agent**, log in as root and use the password you assigned to root.

If you cannot remember your root password, boot your system as **linux single**.

Once you have booted into single user mode and have access to the **#** prompt, you must type **passwd root**, which allows you to enter a new password for root. At this point you can type **shutdown -r now** to reboot the system with the new root password.

If you cannot remember your user account password, you must become root. To become root, type **`su -`** and enter your root password when prompted. Then, type **`passwd <username>`**. This allows you to enter a new password for the specified user account.

If the graphical login screen does not appear, check your hardware for compatibility issues. The *Hardware Compatibility List* can be found at:

```
http://hardware.redhat.com/hcl/
```

14.4.6. Your Printer Does Not Work

If you are not sure how to set up your printer or are having trouble getting it to work properly, try using the **Printer Configuration Tool**.

Type the **`system-config-printer`** command at a shell prompt to launch the **Printer Configuration Tool**. If you are not root, it prompts you for the root password to continue.

14.4.7. Apache-based `httpd` service/Sendmail Hangs During Startup

If you are having trouble with the Apache-based **`httpd`** service or Sendmail hanging at startup, make sure the following line is in the **`/etc/hosts`** file:

```
127.0.0.1  localhost.localdomain  localhost
```

Chapter 15. Additional Boot Options for IBM Power Systems

This section discusses additional boot and kernel boot options available for the Red Hat Enterprise Linux installation program.

To use any of the boot options presented here, type the command you wish to invoke at the installation **boot:** prompt.

Boot Time Command Arguments

askmethod

this command asks you to select the installation method you would like to use when booting from the Red Hat Enterprise Linux CD-ROM.

dd

this argument causes the installation program to prompt you to use a driver diskette.

dd=*url*

this argument causes the installation program to prompt you to use a driver image from a specified HTTP, FTP, or NFS network address.

display=*ip*:0

this command allows remote display forwarding. In this command, *ip* should be replaced with the IP address of the system on which you want the display to appear.

On the system you want the display to appear on, you must execute the command **xhost +*remotehostname***, where *remotehostname* is the name of the host from which you are running the original display. Using the command **xhost +*remotehostname*** limits access to the remote display terminal and does not allow access from anyone or any system not specifically authorized for remote access.

driverdisk

this command performs the same function as the **dd** command and also prompts you to use a driver diskette during the installation of Red Hat Enterprise Linux.

ide=nodma

this command disables DMA on all IDE devices and may be useful when having IDE-related problems.

mediacheck

this command gives you the option of testing the integrity of the install source (if an ISO-based method). this command works with the CD, DVD, hard drive ISO, and NFS ISO installation methods. Verifying that the ISO images are intact before you attempt an installation helps to avoid problems that are often encountered during an installation.

mem=*xxx*m

this command allows you to override the amount of memory the kernel detects for the machine. This may be needed for some older systems where only 16 mb is detected and for some new machines where the video card shares the video memory with the main memory. When executing this command, *xxx* should be replaced with the amount of memory in

megabytes.

`mpath`

enables multipath support.

> **Important — mandatory for installation on multipath devices**
>
> If you install Red Hat Enterprise Linux 5.11 on a network storage device accessible through multiple paths, you must boot the installation process with this option. If you do not specify this option at boot time, installation will fail, or the system will fail to boot after installation completes.

`noeject`

do not eject optical discs after installation. This option is useful in remote installations where it is difficult to close the tray afterwards.

`nopass`

this command disables the passing of keyboard and mouse information to stage 2 of the installation program. It can be used to test keyboard and mouse configuration screens during stage 2 of the installation program when performing a network installation.

`nopcmcia`

this command ignores any PCMCIA controllers in system.

`noprobe`

this command disables hardware detection and instead prompts the user for hardware information.

`noshell`

this command disables shell access on virtual console 2 during an installation.

`nostorage`

this command disables probing for SCSI and RAID storage hardware.

`nousb`

this command disables the loading of USB support during the installation. If the installation program tends to hang early in the process, this command may be helpful.

`nousbstorage`

this command disables the loading of the usbstorage module in the installation program's loader. It may help with device ordering on SCSI systems.

`rescue`

this command runs rescue mode. Refer to Chapter 27, *Basic System Recovery* for more information about rescue mode.

`resolution=`

tells the installation program which video mode to run. it accepts any standard resolution, such as **`640x480`**, **`800x600`**, **`1024x768`**, and so on.

`serial`

this command turns on serial console support.

`text`

this command disables the graphical installation program and forces the installation program to run in text mode.

`updates`

this command prompts you to insert a floppy diskette containing updates (bug fixes) for the anaconda installation program. It is not needed if you are performing a network installation and have already placed the updates image contents in **`rhupdates/`** on the server.

`vnc`

this command allows you to install from a VNC server.

`vncpassword=`

this command sets the password used to connect to the VNC server.

Part III. IBM System z Architecture - Installation and Booting

The *Red Hat Enterprise Linux Installation Guide* for IBM System z Architecture systems discusses the installation of Red Hat Enterprise Linux and some basic post-installation troubleshooting. Advanced installation options are covered in the second part of this manual.

Chapter 16. Steps to Get You Started

16.1. Pre-Installation

The installation process assumes a basic familiarity with the IBM eServer System z platforms. For additional information on these platforms, refer to the IBM Redbooks available online at:

```
http://www.redbooks.ibm.com/
```

This manual assumes you are familiar with the related Redbooks and can set up logical partitions (LPARs) and virtual machines (VMs) on an IBM eServer System z system.

> **Note**
>
> For the most current IBM resources, visit http://www.ibm.com/eserver/zseries/.

Before you install Red Hat Enterprise Linux, you must perform the following steps:

1. Allocate sufficient Disk Storage Space using *DASDs* [8] or *SCSI* [9] partitions to provide suitable disk space (for example, 2 GB is sufficient for server installations, while 5 GB is minimally required to install all packages).

   > **Important — unformatted DASDs on System z**
   >
   > Red Hat Enterprise Linux 5 cannot use unformatted DASDs when installing with kickstart and the cmdline user interface. Refer to the **`clearpart`** documentation in Section 31.4, “Kickstart Options” for a method to ensure that DASDs are formatted during installation.

2. Acquire a minimum of 512 MB RAM (1 GB is strongly recommended) to designate for the Linux virtual machine.

3. Determine if you need swap space and if so how much. While it is possible (and recommended) to assign enough memory to z/VM and let z/VM do the necessary swapping, there may be cases where the amount of required RAM is not predictable. Such instances should be examined on a case-by-case basis.

4. Decide on the environment under which to run the operating system (on an LPAR or as a guest operating system on one or more virtual machines).

5. Finally, it is important to review sections 3.3 through 3.8, and Chapters 5 and 6 of the *IBM Linux for System z Redbook*, as it explains the different configurations and install scenarios available on the zSeries platform as well as how to setup an initial LPAR or Linux virtual machine (z/VM).

16.2. Additional Hardware Preparation for System z

The network configuration must be determined beforehand. Red Hat Enterprise Linux for IBM System z supports QDIO-enabled (Queued Direct I/O) and LCS (LAN channel station) devices. The CTC (channel-to-channel) and IUCV (inter-user communication vehicle) are deprecated and are not supported in Red Hat Enterprise Linux.

For the purposes of this installation, it is recommended that at least 4 GB of disk space (such as two 2 GB DASD, direct access storage device, partitions or equivalent IBM System z SCSI LUNs) be allocated for the installation process. All DASD disk allocations should be completed prior to the installation process. After the installation, more DASD or SCSI (for IBM System z only) disk partitions may be added or deleted as necessary.

16.3. Basic Overview of the Boot Method

To prepare for installation, you must have the Linux kernel (**kernel.img**), the ram disk (**initrd.img**), and if using z/VM, an optional CMS configuration file (**redhat.conf**) and a parameter file. Sample parameter and CMS configuration files are provided (**redhat.parm** and **redhat.conf**). You should edit the CMS configuration file and add information about your DASD. You may also want to add some information about your network configuration. Once this is started on the IBM System z, the networking is configured. You can then use **ssh** on another computer to log into your installation image. Now you can start an installation script to install Red Hat Enterprise Linux.

16.4. Preparing for a Network Installation

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

The Red Hat Enterprise Linux installation media must be available for either a network installation (via NFS, FTP, or HTTP) or installation via local storage. Use the following steps if you are performing an NFS, FTP, or HTTP installation.

The NFS, FTP, or HTTP server to be used for installation over the network must be a separate machine which can provide the complete contents of the installation DVD-ROM or the installation CD-ROMs.

> **Note**
>
> In the following examples, the directory on the installation staging server that will contain the installation files will be specified as ***/location/of/disk/space***. The directory that will be made publicly available via FTP, NFS, or HTTP will be specified as */publicly/available/directory*. For example, ***/location/of/disk/space*** may be a directory you create called **/var/isos**. ***/publicly/available/directory*** might be **/var/www/html/rhel5**, for an HTTP install.

To copy the files from the installation DVD or CD-ROMs to a Linux machine which acts as an installation staging server, perform the following steps:

- Create an iso image from the installation disk(s) using the following command (for DVDs):

  dd if=/dev/*dvd* of=/*location/of/disk/space*/RHEL5.iso

  where *dvd* refers to your DVD drive device.

16.4.1. Preparing for FTP and HTTP installation

If you are setting up an installation tree for NFS, FTP, or HTTP installations, you must copy the **`RELEASE-NOTES`** files and all files from the **`RedHat`** directory on all operating systems ISO images. On Linux and UNIX systems, the following process will properly configure the target directory on your server (repeat for each CD-ROM/ISO image):

1. Insert CD-ROM or DVD-ROM.
2. **`mount /media/cdrom`**
3. If you are installing the Server variant, run **`cp -a /media/cdrom/Server`** ***`<target-directory>`***

   If you are installing the Client variant, run **`cp -a /media/cdrom/Client`** ***`<target-directory>`***
4. **`cp /media/cdrom/RELEASE-NOTES*`** ***`<target-directory>`*** (Installation CD 1 or DVD only)
5. **`cp /media/cdrom/images`** ***`<target-directory>`*** (Installation CD 1 or DVD only)
6. **`umount /media/cdrom`**

(Where ***`<target-directory>`*** represents the path to the directory to contain the installation tree.)

> **Note**
>
> Do *not* copy the Supplementary ISO image, or any of the layered product ISO images, as this will overwrite files necessary for Anaconda's proper operation.
>
> These ISO images must be installed *after* Red Hat Enterprise Linux has been installed.

Next make sure that the ***`/publicly/available/directory`*** directory is shared via FTP or HTTP, and verify client access. You can check to see whether the directory is accessible from the server itself, and then from another machine on the same subnet that you will be installing to.

16.4.2. Preparing for an NFS install

For NFS installation it is not necessary to mount the iso image. It is sufficient to make the iso image itself available via NFS. You can do this by moving the iso image or images to the NFS exported directory:

- For DVD:

  `mv /location/of/disk/space/RHEL5.iso /publicly/available/directory/`

- For CDROMs:

  `mv /location/of/disk/space/disk*.iso /publicly/available/directory/`

Ensure that the ***`/publicly/available/directory`*** directory is exported via NFS via an entry in **`/etc/exports`**.

To export to a specific system:

```
/publicly/available/directory client.ip.address
```

To export to all systems use an entry such as:

```
/publicly/available/directory *
```

Start the NFS daemon (on a Red Hat Enterprise Linux system, use **/sbin/service nfs start**). If NFS is already running, reload the configuration file (on a Red Hat Enterprise Linux system use **/sbin/service nfs reload**).

Be sure to test the NFS share following the directions in the Red Hat Enterprise Linux Deployment Guide.

16.5. Preparing for a Hard Drive Installation

> **Note**
>
> Hard drive installations using DASD or SCSI source storage only work from native ext2 or ext3 partitions. If you have a file system based on devices other than native ext2 or ext3 (particularly a file system based on RAID or LVM partitions) you will not be able to use it as a source to perform a hard drive installation.

Ensure that you have boot media available on CD, DVD, or a USB storage device such as a flash drive.

To prepare your system for a hard drive installation, you must set the system up in one of the following ways:

- Using a set of CD-ROMs, or a DVD — Create ISO image files from each installation CD-ROM, or from the DVD. For each CD-ROM (once for the DVD), execute the following command on a Linux system:

  ```
  dd if=/dev/cdrom of=/tmp/file-name.iso
  ```

  This command may raise an error message when the data at the end of the CD-ROM is reached which can be ignored. The ISO images created can now be used for installation, once copied to the correct DASD.

- Using ISO images - transfer these to the system to be installed (or to the correct DASD or SCSI devices).

  Verifying that ISO images are intact before you attempt an installation, helps to avoid problems. To verify the ISO images are intact prior to performing an installation, use an **md5sum** program (many **md5sum** programs are available for various operating systems). An **md5sum** program should be available on the same Linux machine as the ISO images.

  Make the correct DASDs or SCSI LUNs accessible to the new VM or LPAR, and then proceed with installation.

  Additionally, if a file called **updates.img** exists in the location from which you install, it is used for updates to **anaconda**, the installation program. Refer to the file **install-methods.txt** in the **anaconda** RPM package for detailed information on the various ways to install Red Hat Enterprise Linux, as well as how to apply the installation program updates.

16.6. Installing under z/VM

16.6. Installing under z/VM

Log onto z/VM as the Linux guest account. You can use x3270 or c3270 (from the x3270-text package in Red Hat Enterprise Linux) to log in to z/VM from other Linux systems. Alternatively, use the 3270 terminal emulator on the IBM System z management console. If you are working from a Windows based machine, Jolly Giant (http://www.jollygiant.com) offers an SSL-enabled 3270 emulator.

If you are not in CMS mode, enter it now.

```
i cms
```

If necessary, add the device containing z/VM's TCP/IP tools to your CMS disk list. For example:

```
vmlink tcpmaint 592 592
```

If using any of the qdio/qeth based network connection types (such as *OSA express* or *hipersockets*), set the VM guest qioassist parameter off:

```
set qioassist off
```

FTP to the machine containing the boot images (**kernel.img** and **initrd.img**), log in, and execute the following commands. Use the **(repl** option if you are overwriting existing **kernel.img**, **initrd.img**, **generic.prm**, or **redhat.exec** files:

```
cd /location/of/boot/images//images/
locsite fix 80
bin
get kernel.img (repl
get initrd.img (repl
ascii
get generic.prm (repl
get redhat.exec (repl
quit
```

You may now create the parameter file (for example, **redhat.parm**). Refer to Chapter 19, *Sample Parameter Files* for sample **parm** files. Below is an explanation of the **parm** file contents.

There is a limit of 32 total parameters in the parameter file. In order to accommodate limitations with parameter files, a new configuration file on a CMS DASD should be used to configure the initial network setup and the DASD specification.

A **.parm** file is still required for the *real* kernel parameters, such as ***root=/dev/ram0 ro ip=off ramdisk_size=40000***, and single parameters which are not assigned to variables, such as ***vnc***. Two parameters which are used in z/VM installs to point the installation program at the new CMS configuration file need to be added to the **.parm** file:

```
CMSDASD=191 CMSCONFFILE=redhat.conf
```

CMSDASD is the device ID of the CMS formatted DASD which contains the configuration file. CMSDASD is often the 'A' DASD (usually disk 191) of the z/VM guest account. The name of the configuration file must be set with CMSCONFFILE and needs to be all lowercase.

The syntax of the CMSCONFFILE is bash style with ***variable*="*value*"** pairs, one on each line.

Example **redhat.parm** file:

```
root=/dev/ram0 ro ip=off ramdisk_size=40000
CMSDASD=191 CMSCONFFILE=redhat.conf
vnc
```

The content of the **redhat.exec** file shipped by Red Hat is:

```
/* */
'cl rdr'
'purge rdr all'
'spool punch * rdr'
'PUNCH KERNEL IMG A (NOH'
'PUNCH REDHAT PARM A (NOH'
'PUNCH INITRD IMG A (NOH'
'ch rdr all keep nohold'
'i 00c'
```

Example **redhat.conf** file:

```
HOSTNAME="foobar.systemz.example.com"
DASD="200-203"
NETTYPE="qeth"
IPADDR="192.168.17.115"
SUBCHANNELS="0.0.0600,0.0.0601,0.0.0602"
PORTNAME="FOOBAR"
NETWORK="192.168.17.0"
NETMASK="255.255.255.0"
BROADCAST="192.168.17.255"
SEARCHDNS="example.com:systemz.example.com"
GATEWAY="192.168.17.254"
DNS="192.168.17.1"
MTU="4096"
```

The following parameters are required and must be included in the parameter file:

- **DASD=*dasd-list***

  Where ***dasd-list*** represents the list of DASD devices to be used by Red Hat Enterprise Linux.

  Although autoprobing for DASDs is done if this parameter is omitted, it is highly recommended to include the **DASD=** parameter, as the device numbers (and therefore the device names) can vary when a new DASD is added to the Red Hat Enterprise Linux guest. This can result in an unusable system.

  Additionally, in SAN-based environments, autoprobing in an LPAR-based install may have unintended side effects, as the number of DASD and SCSI volumes visible may be unexpectedly large and include volumes currently in use by other users. In particular, autoprobing during a kickstart install (which may have enabled autopartitioning to clear all partitions) is highly discouraged.

- **root=*file-system***

  where ***file-system*** represents the device on which the root file system can be found. For installation purposes, it should be set to **/dev/ram0**, which is the ramdisk containing the Red Hat Enterprise Linux installation program.

The following parameters are required to set up networking:

- **SUBCHANNELS=**

  Provides required device bus IDs for the various network interfaces.

```
qeth: SUBCHANNELS="read_device_bus_id,write_device_bus_id, data_device_bus_id"
lcs: SUBCHANNELS="read_device_bus_id,write_device_bus_id"
```

For example (a sample qeth SUBCHANNEL statement):

```
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
```

The following parameters are optional:

- **`HOSTNAME=`*`string`***

  Where ***`string`*** is the hostname of the newly-installed Linux guest.

- **`NETTYPE=`*`type`***

  Where ***`type`*** must be one of the following: **`lcs`**, or **`qeth`**.

  Choose **`lcs`** for:

  - OSA-2 Ethernet/Token Ring
  - OSA-Express Fast Ethernet in non-QDIO mode
  - OSA-Express High Speed Token Ring in non-QDIO mode
  - Gigabit Ethernet in non-QDIO mode

  Choose **`qeth`** for:

  - OSA-Express Fast Ethernet
  - Gigabit Ethernet (including 1000Base-T)
  - High Speed Token Ring
  - HiperSockets
  - ATM (running Ethernet LAN emulation)

- **`IPADDR=`*`IP`***

  Where ***`IP`*** is the IP address of the new Linux guest.

- **`NETWORK=`*`network`***

  Where ***`network`*** is the address of your network.

- **`NETMASK=`*`netmask`***

  Where ***`netmask`*** is the netmask.

- **`BROADCAST=`*`broadcast`***

  Where ***`broadcast`*** is the broadcast address.

- **`GATEWAY=`*`gw`***

  Where ***`gw`*** is the gateway-IP for your **`eth`** device.

- **`MTU=`*`mtu`***

Where ***mtu*** is the Maximum Transmission Unit (MTU) for this connection.

- **DNS=*server1*:*server2*::*serverN***

  Where ***server1*:*server2*::*serverN*** is a list of DNS servers, separated by colons. For example:

  DNS=10.0.0.1:10.0.0.2

- **SEARCHDNS=*domain1*:*domain2*::*domainN***

  Where ***domain1*:*domain2*::*domainN*** is a list of the search domains, separated by colons. For example:

  SEARCHDNS=example.com:example.org

- **PORTNAME=*osa_portname* | *lcs_portnumber***

  This variable supports OSA devices operating in qdio mode or non-qdio mode.

  When using qdio mode: *osa_portname* is the portname specified on the OSA device when operating in qeth mode. PORTNAME is only required for z/VM 4.3 or older without APARs VM63308 and PQ73878.

  When using non-qdio mode: *lcs_portnumber* is used to pass the relative port number as integer in the range of 0 through 15.

- **FCP_n="device_number SCSI_ID WWPN SCSI_LUN FCP_LUN"**

  The variables can be used on systems with FCP devices to preconfigure the FCP setup and can be subsequently edited in anaconda during the installation. An example value may look similar to:

  ```
  FCP_1="0.0.5000 0x01 0x5105074308c212e9 0x0 4010"
  ```

  - *n* is an integer value (e.g. **FCP_1**, **FCP_2**, ...).
  - *device_number* is used to specify the address of the FCP device (**0.0.5000** for device 5000, for example).
  - *SCSI_ID* is specified in hex-value, typically sequential values (e.g. **0x01**, **0x02** ...) are used over multiple FCP_ variables.
  - *WWPN* is the world wide port name used for routing (often in conjunction with multipathing) and is as a 16-digit hex value (e.g. **0x5105074308c212e9**).
  - *SCSI_LUN* refers to the local SCSI logical unit value and is specified as a hex-value, typically sequential values (e.g. 0x00, 0x01, ...) are used over multiple FCP_ variables.
  - *FCP_LUN* refers to the storage logical unit identifier and is specified as a hex-value (such as **0x4010**).

  Note

  Each of the values used in the FCP parameters (FCP_1, FCP_2, ...) are site-specific and are normally supplied by the FCP storage administrator.

Update for Red Hat Enterprise Linux 5.3

Anaconda now supports both ports on CHPID for OSA Express3 cards. The installer will prompt for the port number in the initial stage of the installation. The value provided for the port also affects installed network interface startup script. When port 1 is selected, the value "portno=1" is added to OPTIONS parameter of ifcfg-eth* file.

When installing under z/VM, you can add either *PORTNO=0* (to use port 0) or *PORTNO=1* (to use port 1) to the CMS configuration file to avoid being prompted for the mode.

Update for Red Hat Enterprise Linux 5.2

When installing on a System z guest for *Red Hat Enterprise Linux 5.2*, you can make the mode persistent by adding either *LAYER2=0* or *LAYER2=1* to the CMS configuration file. Use *LAYER2=0* when the OSA is in layer 3 mode, and *LAYER2=1* when the OSA is in layer 2 mode.

When specifying *LAYER2=1*, you can also specify *VSWITCH=1* when connecting to a *VSWITCH*, or *VSWITCH=0* when connecting directly to the OSA. If a VSWITCH is not in use, specify the MAC address in the CMS configuration file using the parameter MACADDR=<*MAC address*>.

The following parameters for kickstart installations are optional:

- **`RUNKS=`*`value`***

  Where ***`value`*** is defined as **`1`** if you want to run the installation program in noninteractive (kickstart) mode in the 3270 terminal, or **`0`** otherwise.

- **`cmdline`**

  When **`cmdline`** is specified, 3270 terminal output becomes much more readable, as the installer disables most escape terminal sequences that are applicable to unix-like consoles, but not supported on the 3270 console.

- Make sure that your kickstart file contains all required parameters before you use either of the RUNKS of cmdline options.

If any of the network parameters required to make the network operate correctly are omitted from the **`parm`** file, a prompt appears during the installation boot process.

If you logged off, reconnect and log in using z/VM guest ID you configured for installation. If you are not in CMS mode, enter it now.

```
i cms
```

Use the downloaded REXX script **`redhat.exec`** that contains the commands necessary to IPL the kernel image and start the installation. After having IPLed CMS, enter **`redhat`** on the 3270 console and press the **`Enter`** key to execute this script.

The initial installation start-up script prompts you for information about your networking and DASDs unless you have specified all necessary information in the **`parm`** file.

Once all questions have been answered, you are ready to begin the core installation program, **loader**. To continue with the installation, refer to Chapter 17, *Installing on IBM System z Systems* for further instructions.

16.7. Installing in an LPAR using the Red Hat Enterprise Linux LPAR CD

The following steps must be taken when installing onto an LPAR.

- Log in on the Hardware Master Console (HMC) or the Support Element Workplace (SEW) as a user with sufficient privileges to install a new OS to an LPAR. The SYSPROG user is recommended.
- Select **Images**, then select the LPAR to which you wish to install. Use the arrows in the frame on the right side to navigate to the **CPC Recovery** menu.
- Double-click on **Load from CD-ROM or Server**.
- In the dialog box that follows, select **`Local CD-ROM`** then click **`Continue`**.
- In the dialog that follows, keep the default selection of **`generic.ins`** then click **`Continue`**.
- Skip to Section 16.9, “Installing in an LPAR (Common Steps)” to continue.

16.8. Installing in an LPAR without the Red Hat Enterprise Linux for System z CD-ROMs

- Log in on the Support Element Workplace as a user with sufficient privileges to install a new OS to an LPAR.
- Select **Images**, then select the LPAR you wish to install to.
- Use the arrows in the frame on the right side to navigate to the **CPC Recovery** menu.
- Double-click on **`Load from CD-ROM or Server`**.
- In the dialog box that follows, select **`FTP Source`**, and enter the following information:

  Host Computer:

  Hostname or IP address of the FTP server you wish to install from (for example, **`ftp.redhat.com`**)

  User ID:

  Your user name on the FTP server (or *anonymous*)

  Password:

  Your password (use your email address if you are logging in as anonymous)

  Account:

  Leave this field empty

  File location (can be left blank):

  Directory on the FTP server holding Red Hat Enterprise Linux for System z (for example, **`/pub/redhat/linux/rawhide/s390x`**)

- Click **Continue.**
- In the dialog that follows, keep the default selection of **redhat.ins** and click **Continue**.
- Refer to Section 16.9, “Installing in an LPAR (Common Steps)” to continue.

16.9. Installing in an LPAR (Common Steps)

Once the installation program has started (if the red field behind the LPAR icon is disappearing, the installation program has begun), select the LPAR and double-click on **Operating System Messages**.

The initial installation start-up script asks you questions about your networking and DASD configurations. *Red Hat Enterprise Linux has changed the limit for parameter file definitions and now accepts thirty-two (32) parameters.* Any information not specified in the parameter file must be specified by answering the installation program questions.

Once all questions have been answered, you are ready to begin the core installation program, **loader**. To continue with the installation, refer to Chapter 17, *Installing on IBM System z Systems* for further instructions.

> **Note**
>
> If you install over a network with a Hard Drive Installation source you must perform a text mode installation.

16.10. Do You Have Enough Disk Space?

Nearly every modern-day operating system (OS) uses *disk partitions*, and Red Hat Enterprise Linux is no exception. When you install Red Hat Enterprise Linux, you may have to work with disk partitions.

The disk space used by Red Hat Enterprise Linux must be separate from the disk space used by other OSes you may have installed on your system.

For more information about disks and partition configuration, refer to Section 17.14.4, “Recommended Partitioning Scheme”.

---

[8] Direct Access Storage Devices (or DASDs) are hard disks that allow a maximum of three (3) partitions per DASD. For example, **dasda** has **dasda[123]**.

[9] Using the zFCP driver over fiber and a dedicated switch, SCSI LUNs can be presented to the linux guest as if they were locally attached SCSI drives.

Chapter 17. Installing on IBM System z Systems

This chapter explains how to perform a Red Hat Enterprise Linux installation using the graphical, mouse-based installation program. The following topics are discussed:

- Becoming familiar with the installation program's user interface
- Starting the installation program
- Selecting an installation method
- Configuration steps during the installation (language, keyboard, mouse, partitioning, etc.)
- Finishing the installation

17.1. The Graphical Installation Program User Interface

If you have used a *graphical user interface (GUI)* before, you are already familiar with this process; use your mouse to navigate the screens, click buttons, or enter text fields.

You can also navigate through the installation using the keyboard. The **Tab** key allows you to move around the screen, the Up and Down arrow keys to scroll through lists, **+** and **-** keys expand and collapse lists, while **Space** and **Enter** selects or removes from selection a highlighted item. You can also use the **Alt**+***X*** key command combination as a way of clicking on buttons or making other screen selections, where ***X*** is replaced with any underlined letter appearing within that screen.

> **Note**
>
> While text mode installations are not explicitly documented, those using the text mode installation program can easily follow the GUI installation instructions. One thing to note is that manipulation of LVM (Logical Volume Management) disk volumes and the configuration of zFCP devices are only possible in graphical mode. In text mode it is only possible to view and accept the default LVM setup.

17.2. The Text Mode Installation Program User Interface

The Red Hat Enterprise Linux text mode installation program uses a screen-based interface that includes most of the on-screen *widgets* commonly found on graphical user interfaces. Figure 17.1, “Installation Program Widgets as seen in Disk Druid” illustrates the screen that appears during the installation process.

> **Note**
>
> While text mode installations are not explicitly documented, those using the text mode installation program can easily follow the GUI installation instructions. One thing to note is that manipulation of LVM (Logical Volume Management) disk volumes is only possible in graphical mode. In text mode it is only possible to view and accept the default LVM setup.

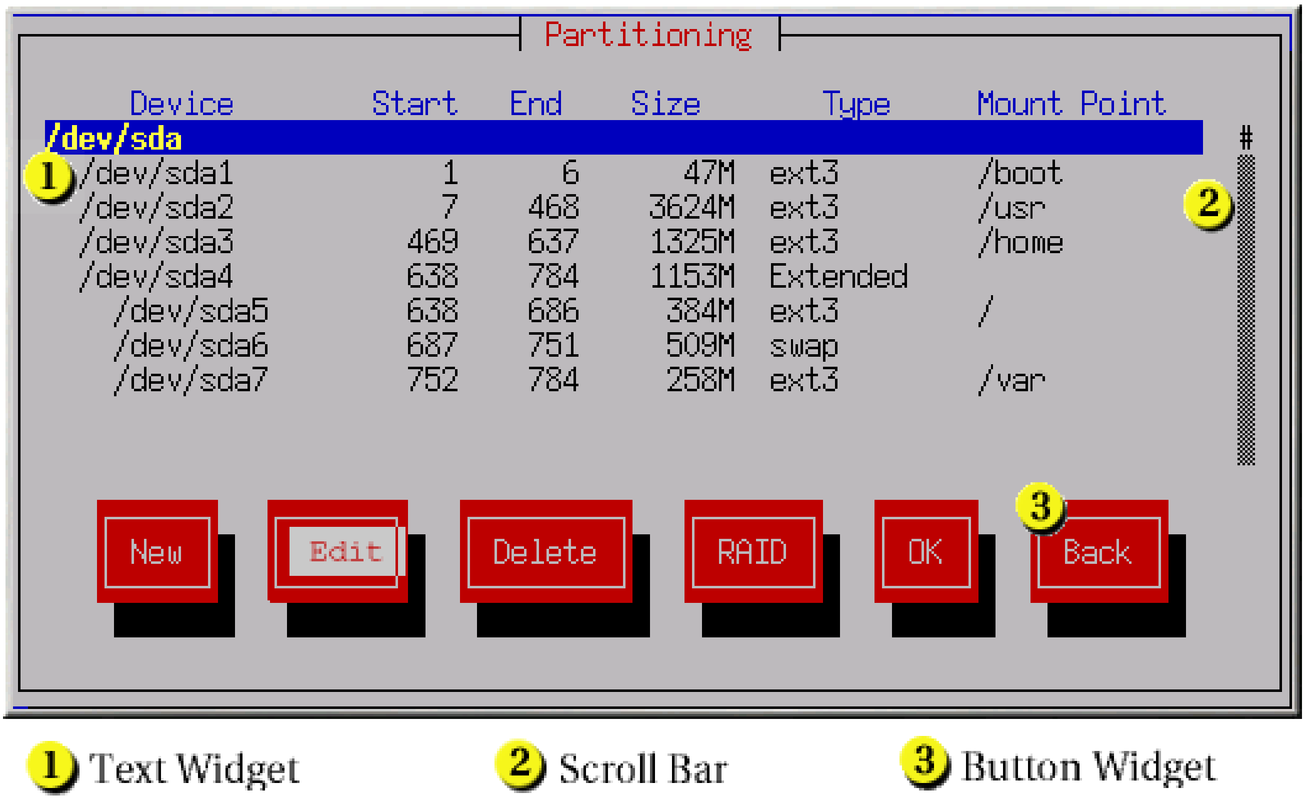

Figure 17.1. Installation Program Widgets as seen in Disk Druid

Here is a list of the most important widgets shown in Figure 17.1, “Installation Program Widgets as seen in Disk Druid”:

- Text Widget — Text widgets are regions of the screen for the display of text. At times, text widgets may also contain other widgets, such as checkboxes. If a text widget contains more information than can be displayed in the space reserved for it, a scroll bar appears; if you position the cursor within the text widget, you can then use the **Up** and **Down** arrow keys to scroll through all the information available. Your current position is shown on the scroll bar by a # character, which moves up and down the scroll bar as you scroll.

- Scroll Bar — Scroll bars appear on the side or bottom of a window to control which part of a list or document is currently in the window's frame. The scroll bar makes it easy to move to any part of a file.

- Button Widget — Button widgets are the primary method of interacting with the installation program. You progress through the windows of the installation program by navigating these buttons, using the **Tab** and **Enter** keys. Buttons can be selected when they are highlighted.

- Cursor — Although not a widget, the cursor is used to select (and interact with) a particular widget. As the cursor is moved from widget to widget, it may cause the widget to change color, or the cursor itself may only appear positioned in or next to the widget. Figure 17.1, “Installation Program Widgets as seen in Disk Druid”, shows the cursor on the **Edit** button.

17.2.1. Using the Keyboard to Navigate

Navigation through the installation dialogs is performed through a simple set of keystrokes. To move the cursor, use the **Left**, **Right**, **Up**, and **Down** arrow keys. Use **Tab**, and **Shift**-**Tab** to cycle forward or backward through each widget on the screen. Along the bottom, most screens display a summary of available cursor positioning keys.

To "press" a button, position the cursor over the button (using **Tab**, for example) and press **Space** or **Enter**. To select an item from a list of items, move the cursor to the item you wish to select and press **Enter**. To select an item with a checkbox, move the cursor to the checkbox and press **Space** to select an item. To deselect, press **Space** a second time.

Pressing **F12** accepts the current values and proceeds to the next dialog; it is equivalent to pressing the **OK** button.

Warning

Unless a dialog box is waiting for your input, do not press any keys during the installation process (doing so may result in unpredictable behavior).

17.3. Running the Installation Program

After following the steps outlined in Chapter 16, *Steps to Get You Started* for booting an LPAR or VM system, **ssh** to the configured Linux install system on the IBM System z.

Although the text mode installation program is run by default for most installations, you can optionally run the graphical installation program available for both VM and LPAR installations via the NFS installation method.

Note

If you have a slow network connection or prefer a text-based installation, do not set the **DISPLAY=** variable in the **parm** file. The text-based installation is similar to the graphical installation; however, the graphical installation offers more package selection details and other options not available in text-based installs. It is strongly recommended to use the graphical installation whenever possible.

To run the graphical installation, use a workstation that has an X Window System server or VNC client installed. Use an SSH client that allows X11 forwarding or a Telnet client. SSH is strongly recommended for its security features as well as its ability to forward X and VNC sessions. Enable X11 forwarding in your SSH client prior to connecting to the *Linux image* (the Linux guest running on z/VM).

17.3.1. Installation using X11 Forwarding

For example, to connect to the Linux image and display the graphical installation program using OpenSSH with X11 forwarding on a Linux workstation, type the following at the workstation shell prompt:

```
ssh -X linuxvm.example.com
```

The **-X** option enables X11 forwarding.

The graphical installation program cannot be started if your DNS or hostnames are not set correctly, or the Linux image is not allowed to open applications on your display. You can prevent this by setting a correct **DISPLAY=** variable. Add the parameter **DISPLAY=*workstationname*:0.0** in the parameter file, replacing *workstationname* with the hostname of the client workstation connecting to

the Linux Image. Allow the Linux image to connect to the workstation using the command **xhost +linuxvm** on the local workstation.

If the graphical installation via NFS does not automatically begin for you, verify the **DISPLAY=** variable settings in the **parm** file. If performing a VM installation, rerun the installation to load the new **parm** file on the reader. Additionally, make sure when performing an X11 forwarded display that the X server is started on the workstation machine. Finally, make sure either the NFS, FTP or HTTP protocols are selected, as all 3 methods support graphical installations.

17.3.2. Installation using VNC

If you are using VNC, a message on the workstation SSH terminal prompts you to start the VNC client viewer and details the VNC display specifications. Enter the specifications from the SSH terminal into the VNC client viewer and connect to the Linux image to begin the installation.

Once you have logged into the Linux image the **loader** will start the installation program.

When the **loader** starts, several screens appear for selecting the installation method.

17.4. Installing from a Hard Drive (DASD)

The **Select Partition** screen applies only if you are installing from a disk partition (that is, if you used the **askmethod** boot options and selected **Hard Drive** in the **Installation Method** dialog). This dialog allows you to name the disk partition and directory from which you are installing Red Hat Enterprise Linux. If you used the **repo=hd** boot option, you already specified a partition.

Enter the device name of the partition containing the Red Hat Enterprise Linux ISO images. This partition must be formatted with a ext2 or vfat filesystem, and cannot be a logical volume. There is also a field labeled **Directory holding images**.

If the ISO images are in the root (top-level) directory of a partition, enter a **/**. If the ISO images are located in a subdirectory of a mounted partition, enter the name of the directory holding the ISO images within that partition. For example, if the partition on which the ISO images is normally mounted as **/home/**, and the images are in **/home/new/**, you would enter **/new/**.

After you have identified the disk partition, the **Welcome** dialog appears.

17.5. Installing via NFS

The NFS dialog applies only if you are installing from an NFS server (if you selected **NFS Image** in the **Installation Method** dialog).

Enter the domain name or IP address of your NFS server. For example, if you are installing from a host named **eastcoast** in the domain **example.com**, enter **eastcoast.example.com** in the **NFS Server** field.

Next, enter the name of the exported directory. If you followed the setup described in Section 16.4, “Preparing for a Network Installation”, you would enter the directory ***/export/directory/***.

If the NFS server is exporting a mirror of the Red Hat Enterprise Linux installation tree, enter the directory which contains the root of the installation tree. You will enter an Installation Key later on in the process which will determine which subdirectories are used to install from. If everything was specified properly, a message appears indicating that the installation program for Red Hat Enterprise Linux is running.

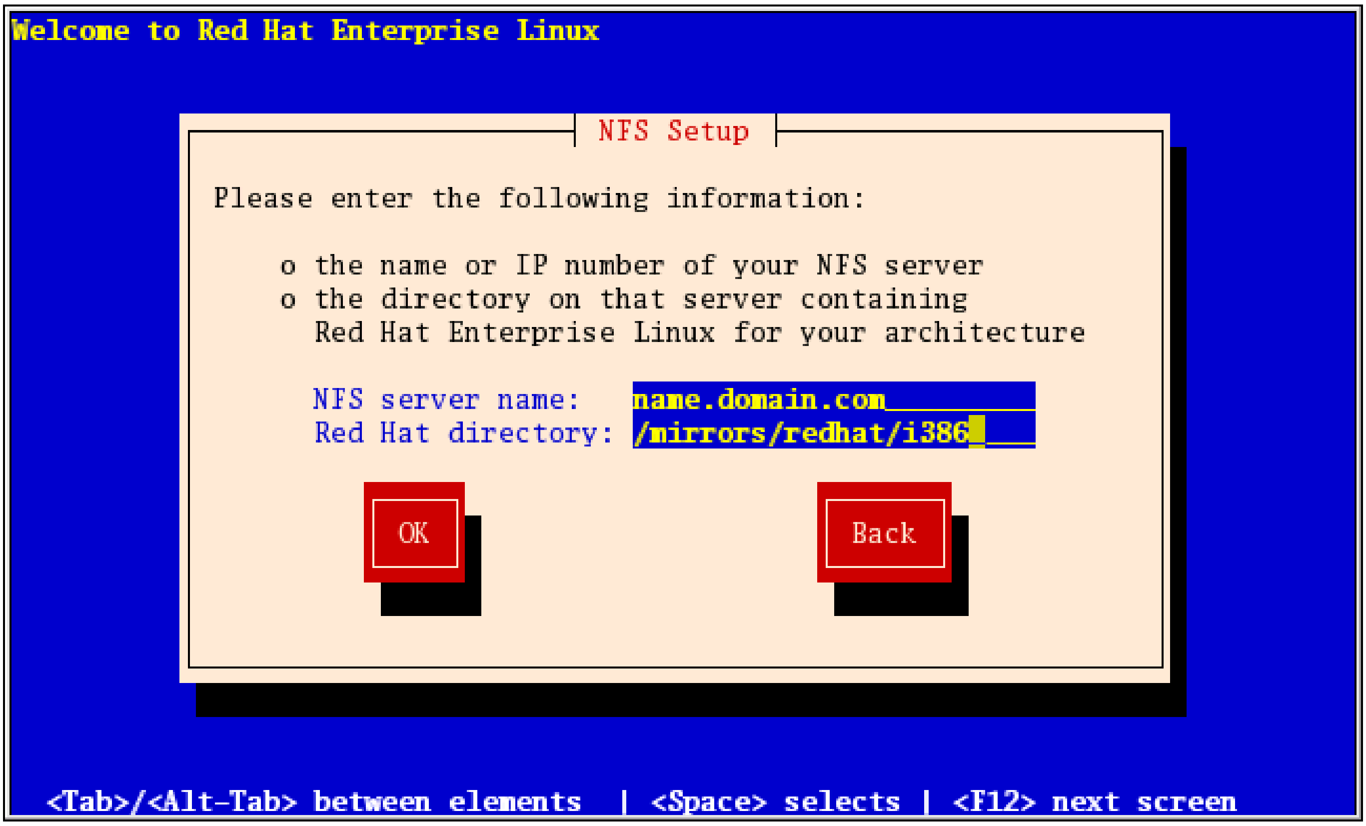

Figure 17.2. NFS Setup Dialog

If the NFS server is exporting the ISO images of the Red Hat Enterprise Linux CD-ROMs, enter the directory which contains the ISO images.

Next, the **`Welcome`** dialog appears.

17.6. Installing via FTP

The FTP dialog applies only if you are installing from an FTP server (that is, if you used the **`askmethod`** boot options and selected **`FTP`** in the **`Installation Method`** dialog). This dialog allows you to identify the FTP server from which you are installing Red Hat Enterprise Linux. If you used the **`repo=ftp`** boot option, you already specified a server and path.

Figure 17.3. FTP Setup Dialog

Enter the name or IP address of the FTP site you are installing from, and the name of the directory containing the ***`variant/`*** directory for your architecture. For example, if the FTP site contains the directory **`/mirrors/redhat/`*`arch`*`/`*`variant`*`;/`**, enter **`/mirrors/redhat/`*`arch`*`/`** (where *arch* is replaced with the architecture type of your system, such as i386, ia64, ppc, or s390x, and *variant* is the variant that you are installing, such as Client, Server, Workstation, etc.). If everything was specified properly, a message box appears indicating that files are being retrieved from the server.

Next, the **`Welcome`** dialog appears.

Note

You can save disk space by using the ISO images you have already copied to the server. To accomplish this, install Red Hat Enterprise Linux using ISO images without copying them into a single tree by loopback mounting them. For each ISO image:

`mkdir disc`*`X`*

`mount -o loop RHEL5-disc`*`X`*`.iso disc`*`X`*

Replace **X** with the corresponding disc number.

17.7. Installing via HTTP

The HTTP dialog applies only if you are installing from an HTTP server (that is, if you used the **`askmethod`** boot option and selected **`HTTP`** in the **`Installation Method`** dialog). This dialog prompts you for information about the HTTP server from which you are installing Red Hat Enterprise Linux. If you used the **`repo=http`** boot option, you already specified a server and path.

Enter the name or IP address of the HTTP site you are installing from, and the name of the directory containing the ***`variant/`*** directory for your architecture. For example, if the HTTP site contains the directory **`/mirrors/redhat/`*`arch`*`/`*`variant`*`/`**, enter **`/mirrors/redhat/`*`arch`*`/`** (where *arch* is replaced with the architecture type of your system, such as i386, ia64, ppc, or s390x, and *variant* is the variant that you are installing, such as Client, Server, Workstation, etc.). If everything was specified properly, a message box appears indicating that files are being retrieved from the server.

Figure 17.4. HTTP Setup Dialog

Next, the **`Welcome`** dialog appears.

Note

You can save disk space by using the ISO images you have already copied to the server. To accomplish this, install Red Hat Enterprise Linux using ISO images without copying them into a single tree by loopback mounting them. For each ISO image:

`mkdir disc`*`X`*

`mount -o loop RHEL5-disc`*`X`*`.iso disc`*`X`*

Replace **X** with the corresponding disc number.

17.8. Welcome to Red Hat Enterprise Linux

The **Welcome** screen does not prompt you for any input. From this screen you can access the Release Notes for Red Hat Enterprise Linux 5.11 by clicking on the Release Notes button.

Click on the **Next** button to continue.

17.9. Language Selection

Using your mouse, select a language to use for the installation (refer to Figure 17.5, “Language Selection”).

The language you select here will become the default language for the operating system once it is installed. Selecting the appropriate language also helps target your time zone configuration later in the installation. The installation program tries to define the appropriate time zone based on what you specify on this screen.

Figure 17.5. Language Selection

Once you select the appropriate language, click **Next** to continue.

17.10. Enter the Installation Number

Enter your Installation Number (refer to Figure 17.6, "Installation Number"). This number will determine the package selection set that is available to the installer. If you choose to skip entering the installation number you will be presented with a basic selection of packages to install later on.

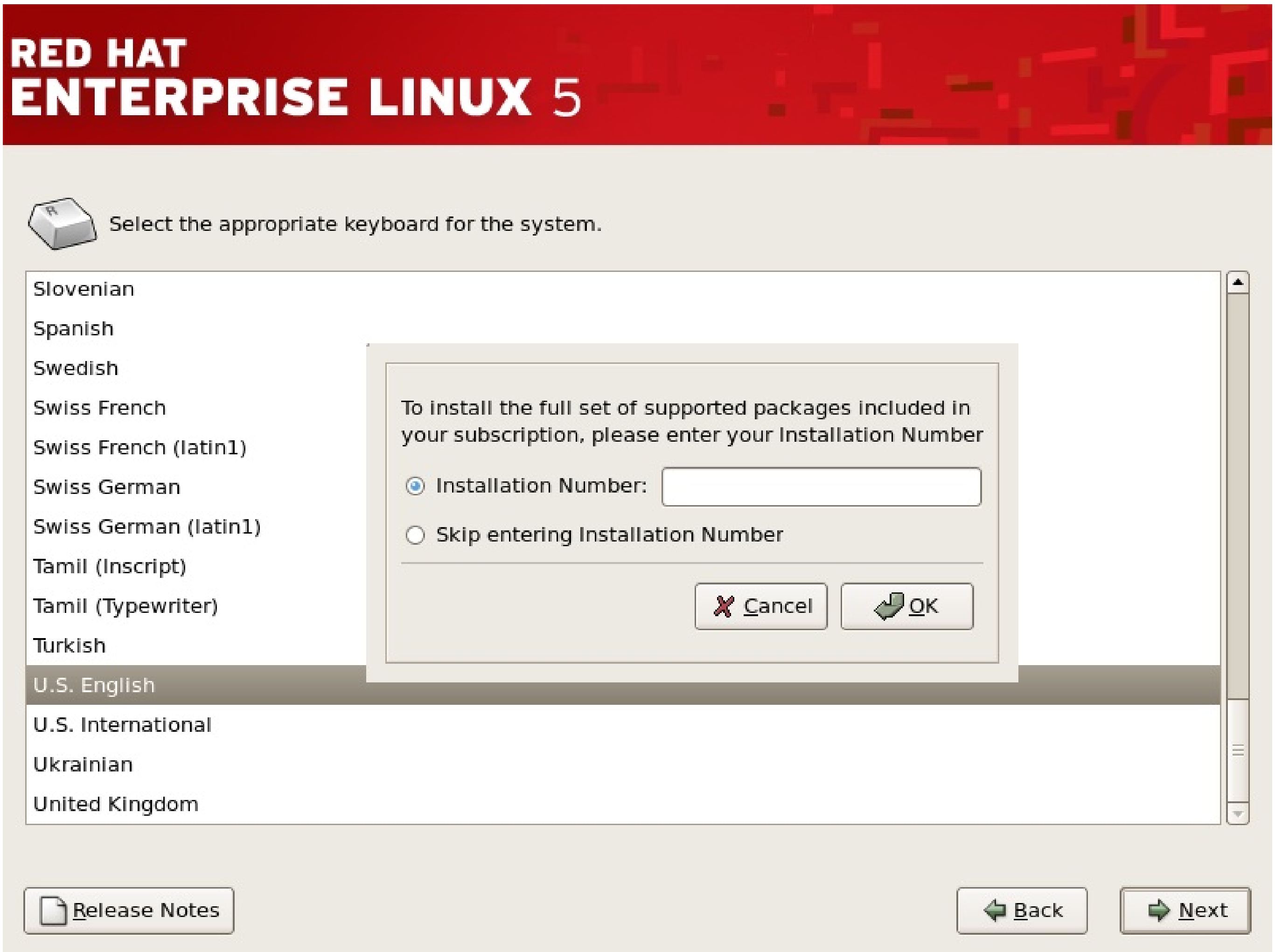

Figure 17.6. Installation Number

17.11. Disk Partitioning Setup

Partitioning allows you to divide your storage drive(s) into isolated sections, where each section behaves as its own drive. Partitioning is particularly useful if you run multiple operating systems, or wish to enforce a logical or functional distinction between your storage partitions (such as a **`/home`** partition that persistently contains user information).

On this screen you can choose to create the default layout or choose to manual partition using the 'Create custom layout' option of **Disk Druid**.

The first three options allow you to perform an automated installation without having to partition your drive(s) yourself. If you do not feel comfortable with partitioning your system, it is recommended that you *do not* choose to create a custom layout and instead let the installation program partition for you.

You can configure an zFCP LUN for installation, or disable a dmraid device from this screen by clicking on the 'Advanced storage configuration' button. For more information refer to Section 17.12, “ Advanced Storage Options ”.

Warning — vdisks, tdisks, and default layout

The default layout uses all devices made available to the installer. This includes any temporary storage devices such as vdisks or tdisks. If temporary storage devices are used with the default layout, the system might become inaccessible when these devices are redefined and the device contents are lost (for example, on logoff and relogon of z/VM guests). Review the partition layout and if temporary devices are included, go back and choose **`Create custom layout`** to exclude these devices from being used for permanent system data.

Warning

The **Update Agent** downloads updated packages to **`/var/cache/yum/`** by default. If you partition the system manually, and create a separate **`/var/`** partition, be sure to create the partition large enough (3.0 GB or more) to download package updates.

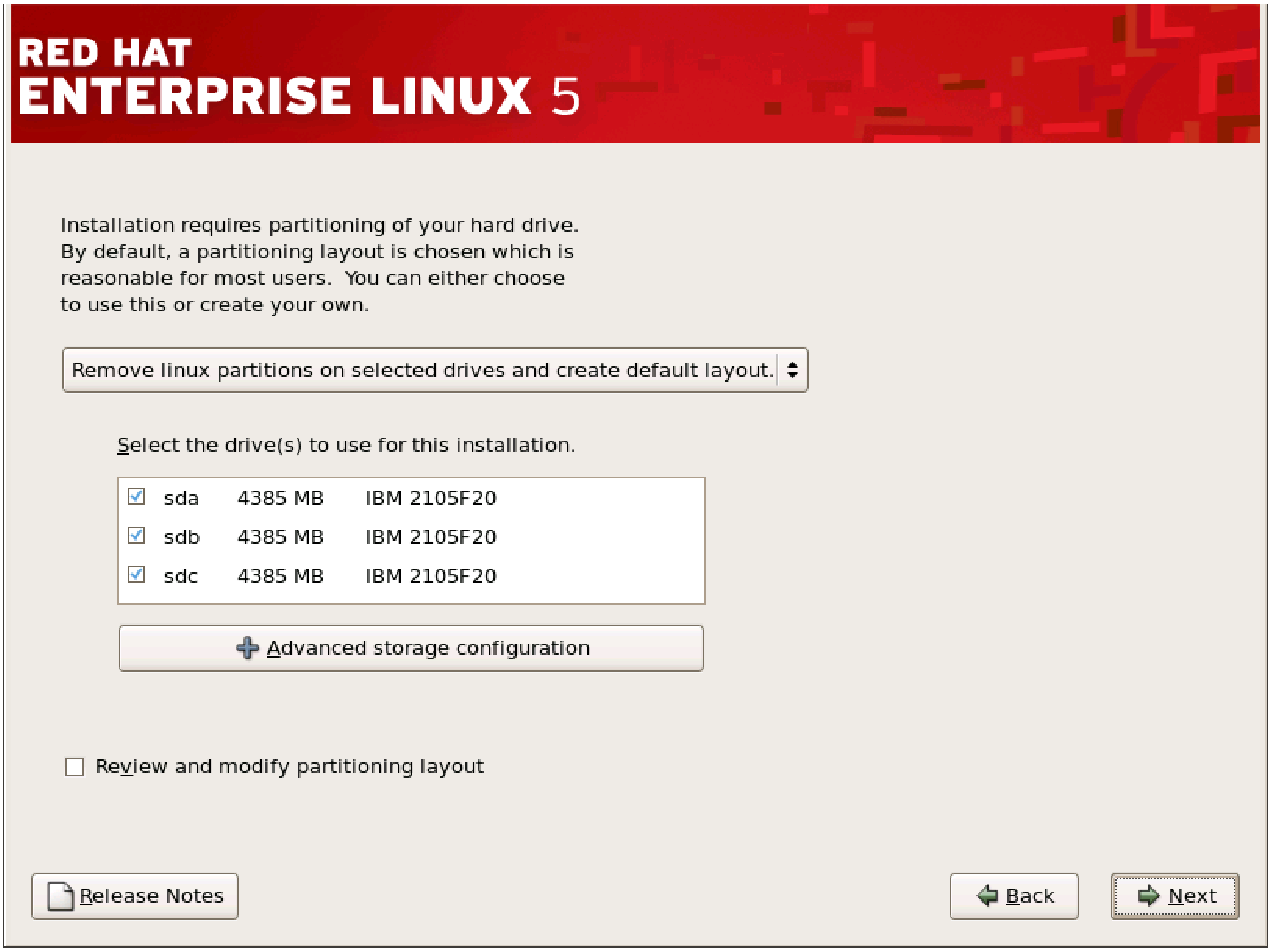

Figure 17.7. Disk Partitioning Setup

If you choose to create a custom layout using **Disk Druid**, refer to Section 17.14, “Partitioning Your System”.

Warning

If you receive an error after the **`Disk Partitioning Setup`** phase of the installation saying something similar to:

"The partition table on device dasda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive."

you may not have a partition table on that drive or the partition table on the drive may not be recognizable by the partitioning software used in the installation program.

No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

Important — multipath devices

To install Red Hat Enterprise Linux on a network device accessible through multiple paths, deselect all local storage in the **`Select the drive(s) to use for this installation`** window, and select a device labeled **`mapper/mpath`** instead.

Note that migrating the root file system of an existing Red Hat Enterprise Linux installation from single path storage to multipath storage is not supported. You must perform a new installation to move the root file system to a multipath storage device. Therefore you should plan your installation accordingly. See https://access.redhat.com/site/solutions/66501 for more information.

17.12. Advanced Storage Options

17.12.1. FCP Devices

FCP (*Fibre Channel protocol*) devices enable IBM System z to use SCSI devices rather than DASD devices. *FCP* (*Fibre Channel protocol*) devices provide a switched fabric topology that enables zSeries systems to use SCSI LUNs as disk devices in addition to traditional DASD devices.

Typically, an operating system is loaded, and the automatic probing and defining of hardware is done by the OS. However, because of the flexibility of configurations associated with FCP, IBM System z requires that any FCP (Fibre Channel protocol) device be entered manually (either in the installation program interactively, or specified as unique parameter entries in the CMS conf file) in order for the installation program to recognize the hardware. The values entered here are unique to each site in which they are setup.

Note

Interactive creation of a zFCP device is only possible in the graphical mode installer. It is not possible to interactively configure a zFCP device in a text-only install.

Each value entered should be verified as correct, as any mistakes made may cause the system not to operate properly.

For more information on these values, refer to the hardware documentation that came with your system and check with the system administrator who has setup the network for this system.

Figure 17.8. Advanced Storage Options

To configure a Fiber Channel Protocol SCSI device invoke the 'Add FCP device' dialog by selecting 'Add ZFCP LUN' and clicking on the 'Add Drive' button. Fill in the details for the 16 bit device number, 64 bit World Wide Port Number (WWPN) and 64 bit FCP LUN. Click the 'Add' button to attempt connection to the FCP device using this information.

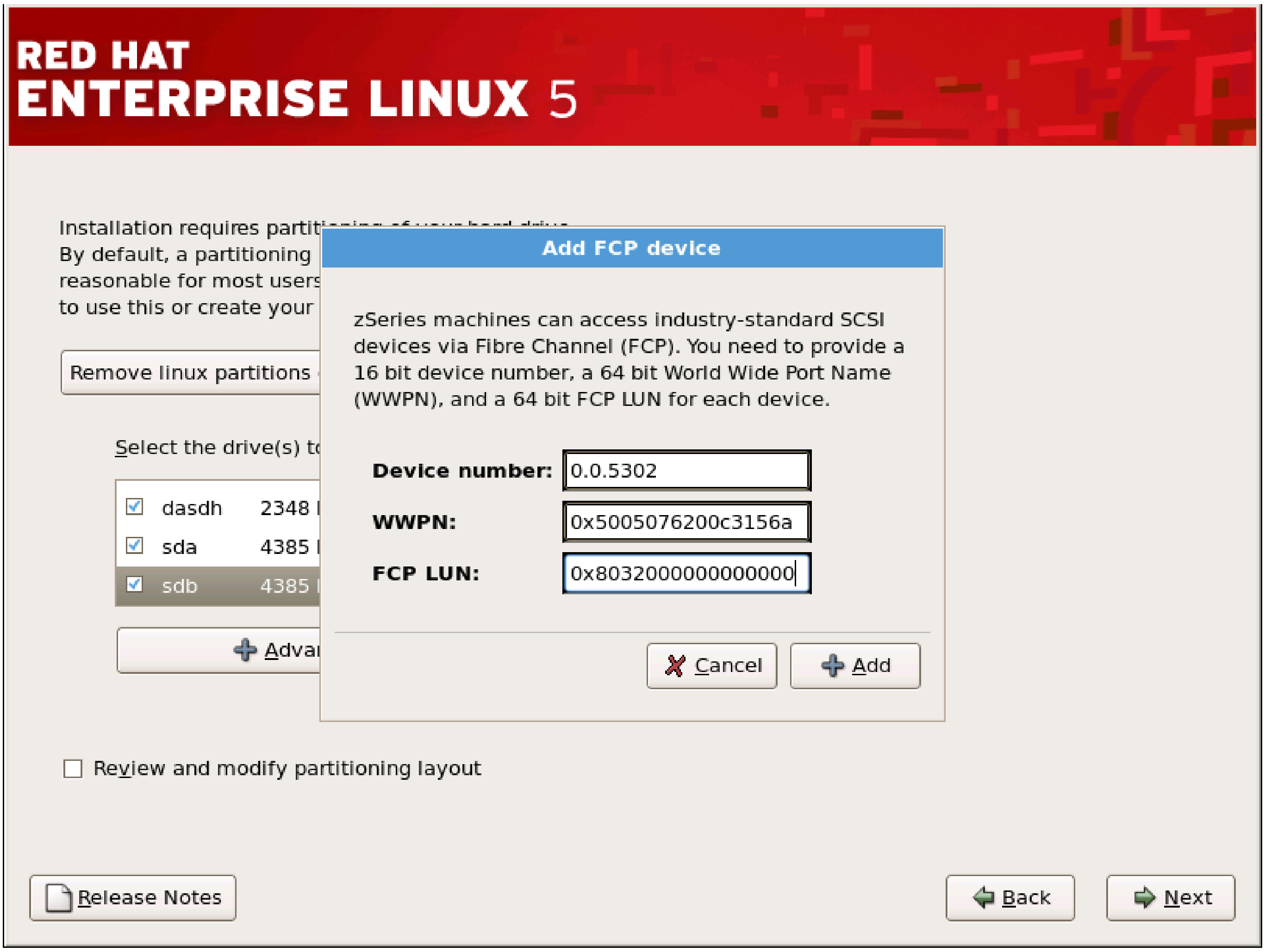

Figure 17.9. Configure FCP Device

The newly added device should then be present and usable during the *Disk Druid* portion of the installation.

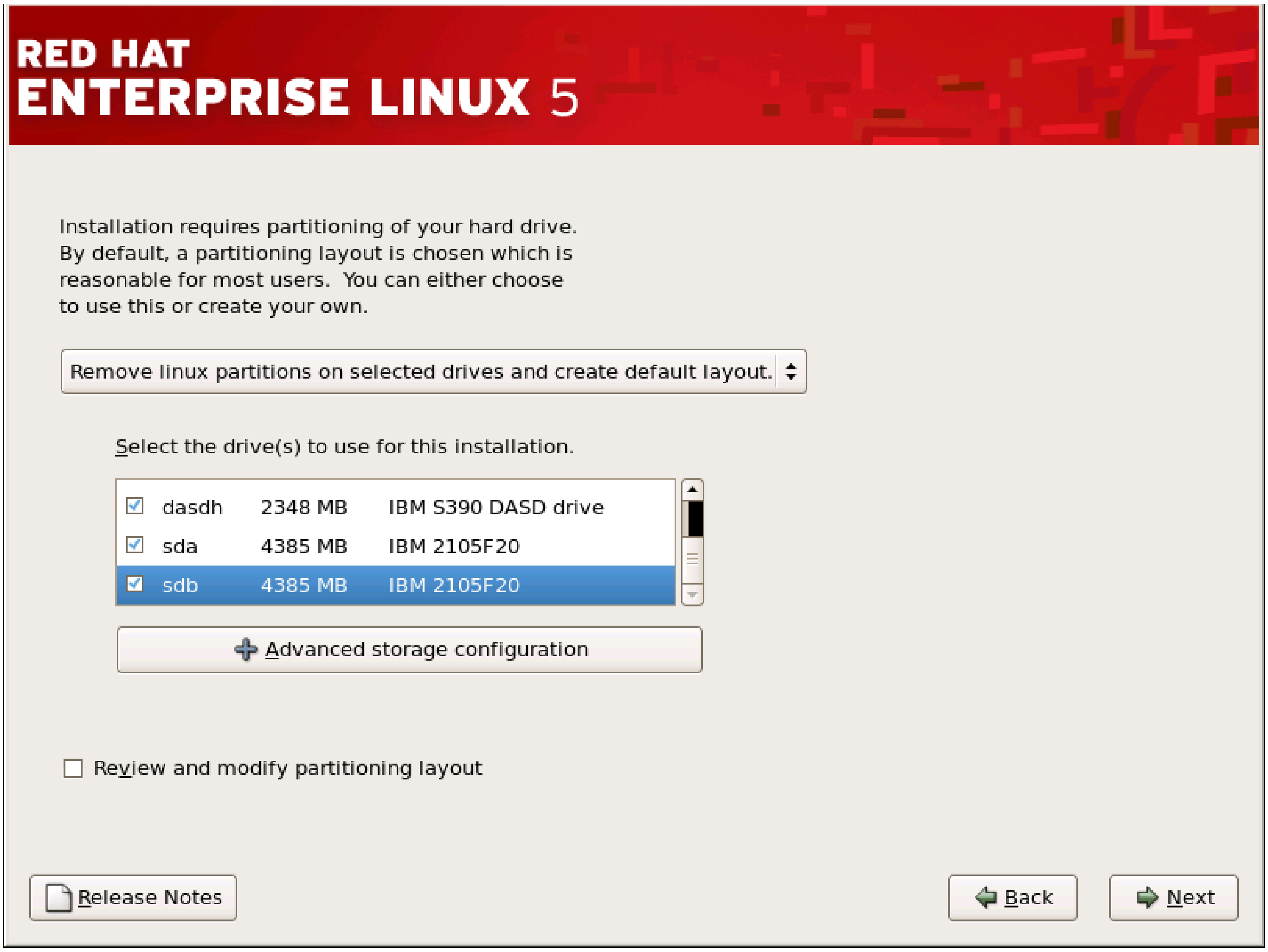

Figure 17.10. Configure FCP Device

Note

The installer requires that at least one ECKD DASD be defined. In the situation where a SCSI-only installation is desired a DASD= parameter should be entered into the CMS conf file with a non-existent device number. This will satisfy Anaconda's requirement for a defined ECKD DASD, while resulting in a SCSI-only environment.

17.13. Create Default Layout

Create default layout allows you to have some control concerning what data is removed (if any) from your system. Your options are:

- **`Remove all partitions on selected drives and create default layout`** — select this option to remove all partitions on your hard drive(s) (this includes partitions created by other operating systems such as z/VM or z/OS).

> **Warning**
>
> If you select this option, all data on the selected DASD and SCSI storage drive(s) is removed by the installation program. Do not select this option if you have information that you want to keep on the storage drive(s) where you are installing Red Hat Enterprise Linux.

- **`Remove Linux partitions on selected drives and create default layout`** — select this option to remove only Linux partitions (partitions created from a previous Linux installation). This does not remove other partitions you may have on your storage drive(s) (such as z/VM or z/OS partitions).
- **`Use free space on selected drives and create default layout`** — select this option to retain your current data and partitions, assuming you have enough free space available on your storage drive(s).

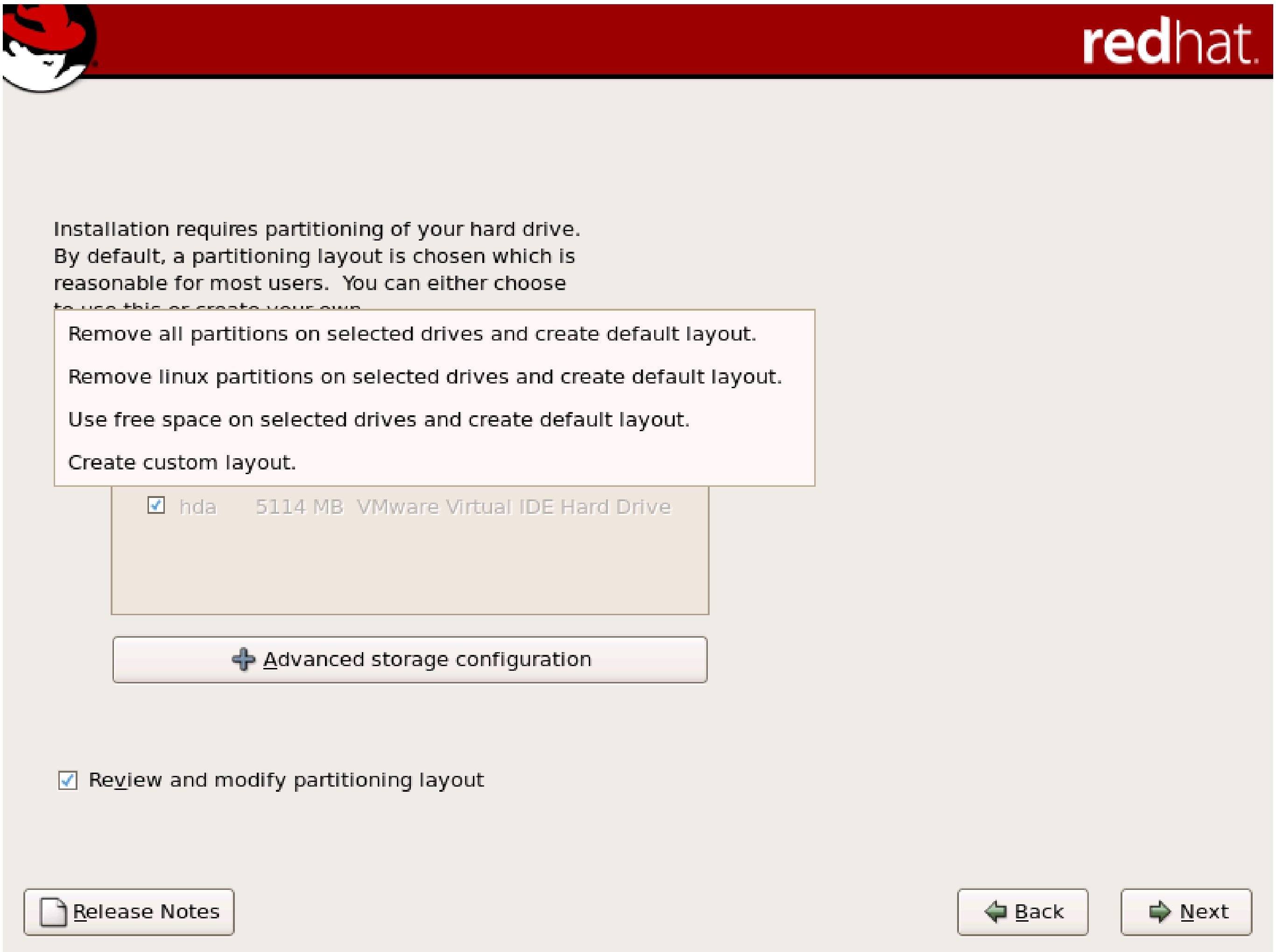

Figure 17.11. Create Default Layout

Using your mouse, choose the storage drive(s) on which you want Red Hat Enterprise Linux to be installed. If you have two or more drives, you can choose which drive(s) should contain this installation. Unselected drives, and any data on them, are not touched.

Warning

It is always a good idea to back up any data that you have on your systems. For example, if you are upgrading or creating a dual-boot system, you should back up any data you wish to keep on your drive(s). Mistakes do happen and can result in the loss of all your data.

To review and make any necessary changes to the partitions created by automatic partitioning, select the **Review** option. After selecting **Review** and clicking **Next** to move forward, the partitions created for you in **Disk Druid** appear. You can make modifications to these partitions if they do not meet your needs.

Click **Next** once you have made your selections to proceed.

17.14. Partitioning Your System

If you chose to create a custom layout, you must tell the installation program where to install Red Hat Enterprise Linux. This is done by defining mount points for one or more disk partitions in which Red Hat Enterprise Linux is installed.

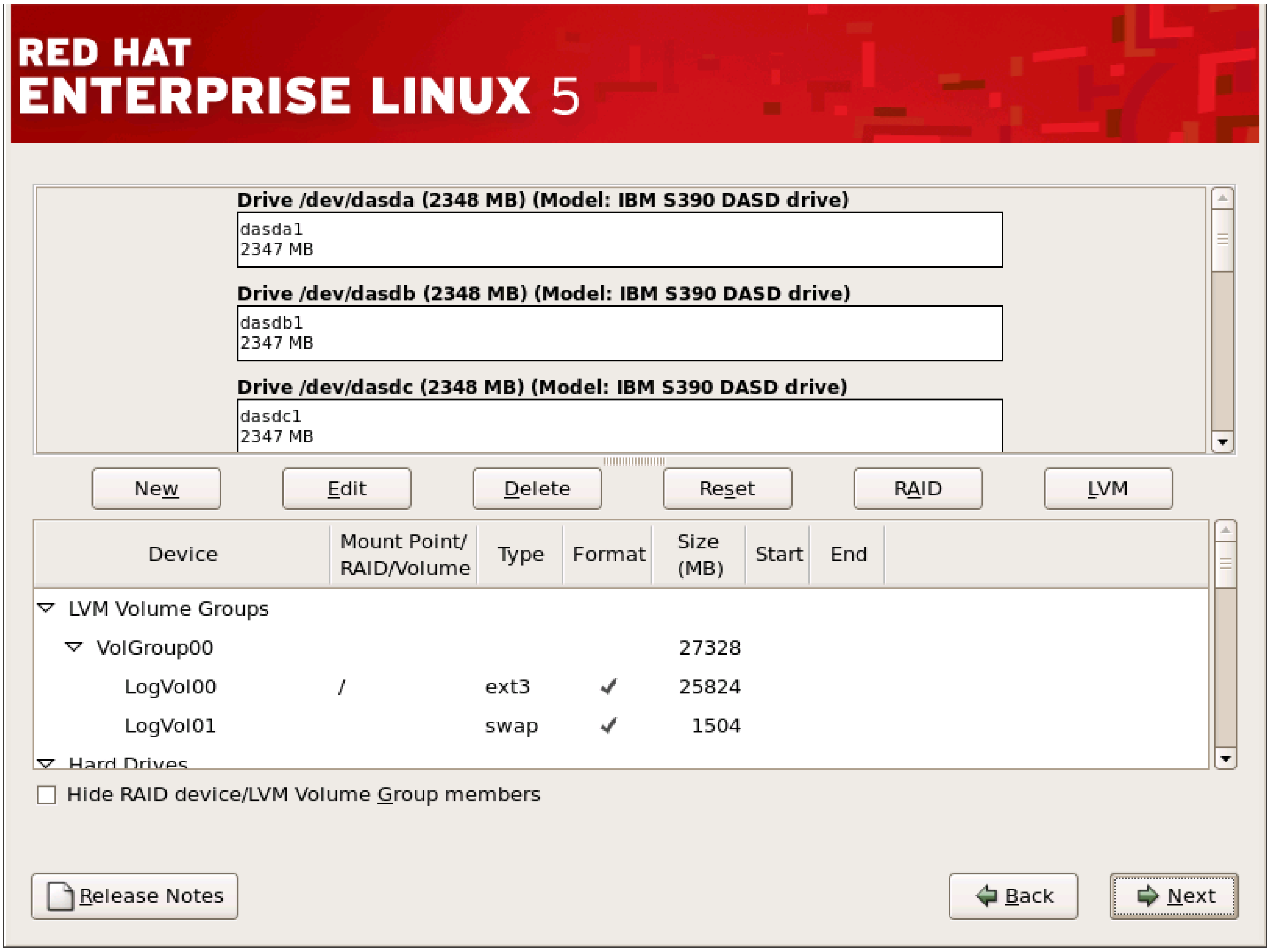

Figure 17.12. Partitioning with Disk Druid

The partitioning tool used by the installation program is **Disk Druid**. With the exception of certain esoteric situations, **Disk Druid** can handle the partitioning requirements for a typical installation.

17.14.1. Graphical Display of DASD Device(s)

Disk Druid offers a graphical representation of your DASD device(s).

Using your mouse, click once to highlight a particular field in the graphical display. Double-click to edit an existing partition or to create a partition out of existing free space.

Above the display, you can review the **`Drive`** name (such as /dev/dasda), the **`Geom`** (which shows the hard disk's geometry and consists of three numbers representing the number of cylinders, heads, and sectors as reported by the hard disk), and the **`Model`** of the hard drive as detected by the installation program.

Finally, note which device is associated with **`/boot`**. The kernel files and bootloader sector will be associated with this device. For most common cases, the first DASD or SCSI LUN will be used, but for some unusual cases, this may not be the case. The device number will be used when re-ipling the post-installed system.

17.14.2. Disk Druid's Buttons

These buttons control **Disk Druid's** actions. They are used to change the attributes of a partition (for example the file system type and mount point) and also to create RAID devices. Buttons on this screen are also used to accept the changes you have made, or to exit **Disk Druid**. For further explanation, take a look at each button in order:

- **`Edit`**: Used to modify attributes of the partition currently selected in the **`Partitions`** section. Selecting **`Edit`** opens a dialog box. Some or all of the fields can be edited, depending on whether the partition information has already been written to disk.

- **`RAID`**: Used to provide redundancy to any or all disk partitions. *It should only be used if you have experience using RAID.* To read more about RAID, see the Red Hat Enterprise Linux Deployment Guide.

  To make a RAID device, you must first create software RAID partitions. Once you have created two or more software RAID partitions, select **`RAID`** to join the software RAID partitions into a RAID device.

17.14.3. Partition Fields

Above the partition hierarchy are labels which present information about the partitions you are creating. The labels are defined as follows:

- **`Device`**: This field displays the partition's device name.

- **`Mount Point/RAID/Volume`**: A mount point is the location within the directory hierarchy at which a volume exists; the volume is "mounted" at this location. This field indicates where the partition is mounted. If a partition exists, but is not set, then you need to define its mount point. Double-click on the partition or click the **`Edit`** button.

- **`Type`**: This field shows the partition's file system type (for example, ext2, ext3, or vfat).

- **`Format`**: This field shows if the partition being created will be formatted.

- **`Size (MB)`**: This field shows the partition's size (in MB).

- **`Start`**: This field shows the cylinder on your hard drive where the partition begins.

- **`End`**: This field shows the cylinder on your hard drive where the partition ends.

`Hide RAID device/LVM Volume Group members`: Select this option if you do not want to view any RAID device or LVM Volume Group members that have been created.

17.14.4. Recommended Partitioning Scheme

Configuring efficient swap space for Linux on System z is a complex task. It very much depends on the specific environment and should be tuned to the actual system load.

Refer to the following resources for more information and to guide your decision:

- 'Chapter 7. Linux Swapping' in the IBM Redbook *Linux on IBM System z: Performance Measurement and Tuning* [IBM Form Number SG24-6926-01], [ISBN 0738485586], available from http://www.redbooks.ibm.com/abstracts/sg246926.html
- *Linux on System z performance* in the IBM Systems Information Center at http://publib.boulder.ibm.com/infocenter/systems/index.jsp?topic=/liaag/lcon_Linux_on_System_z_performance.htm
- *Linux Performance when running under VM*, available from http://www.vm.ibm.com/perf/tips/linuxper.html

17.14.5. Editing Partitions

To edit a partition, select the **`Edit`** button or double-click on the existing partition.

> **Note**
>
> If the partition already exists on your disk, you can only change the partition's mount point. To make any other changes, you must delete the partition and recreate it.

17.15. Network Configuration

If you do not have a network device, this screen does not appear during your installation and you should advance to Section 17.16, “Time Zone Configuration”.

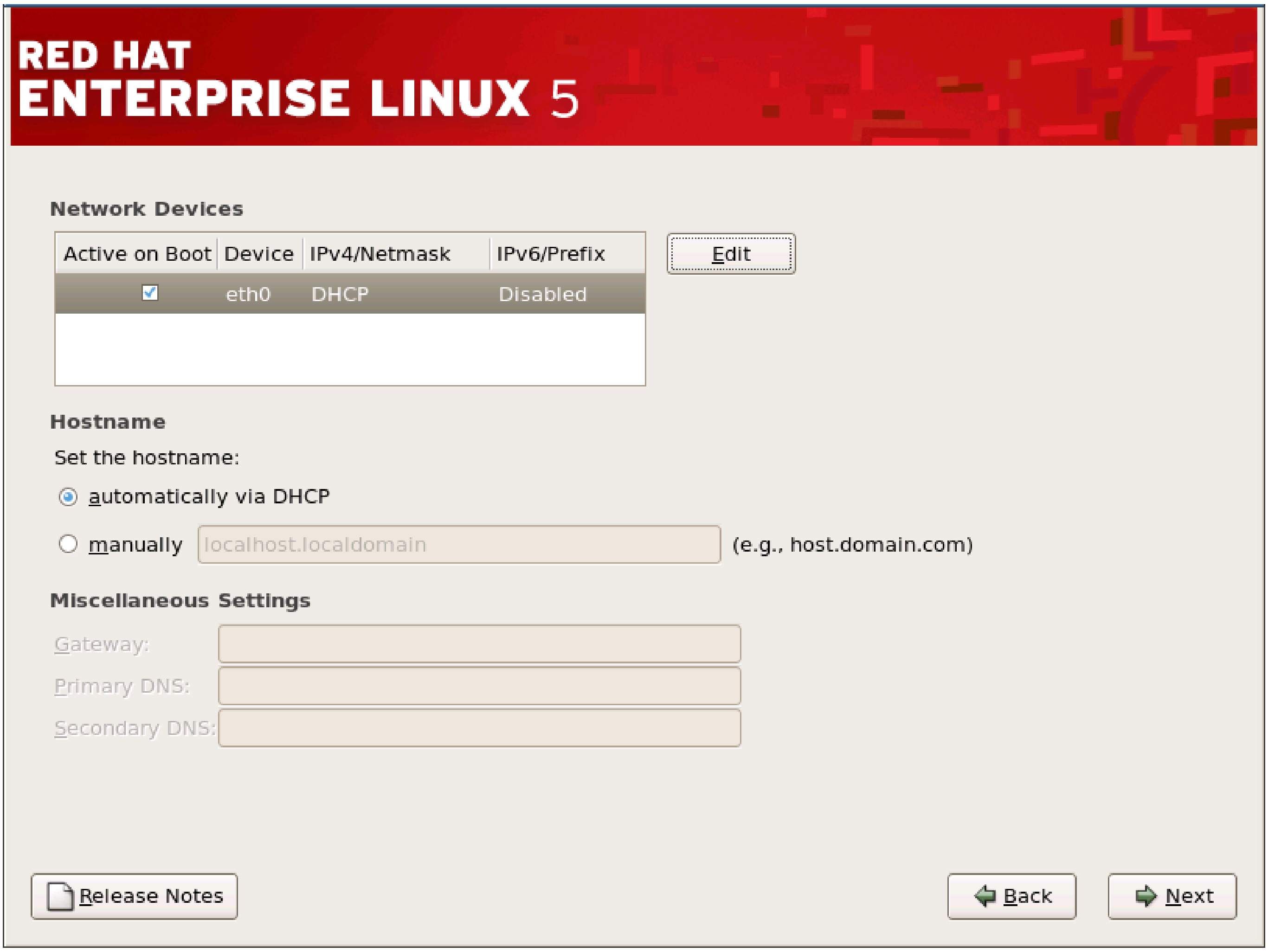

Figure 17.13. Network Configuration

The installation program automatically detects any network devices you have and displays them in the **`Network Devices`** list.

When you have selected a network device, click **`Edit`**. From the **`Edit Interface`** dialog, you can choose to configure the IP address and Netmask (for IPv4 - Prefix for IPv6) of the device to use DHCP or to use static settings. If you do not have DHCP client access or you are unsure what to provide here, contact your network administrator.

Note

DHCP should not be chosen for **`qdio/qeth`** devices that are configured with OSA layer 3 support. Layer 3 does not provide MAC address or Address Resolution Protocol (ARP) abilities and so can not be used with network services that require them.

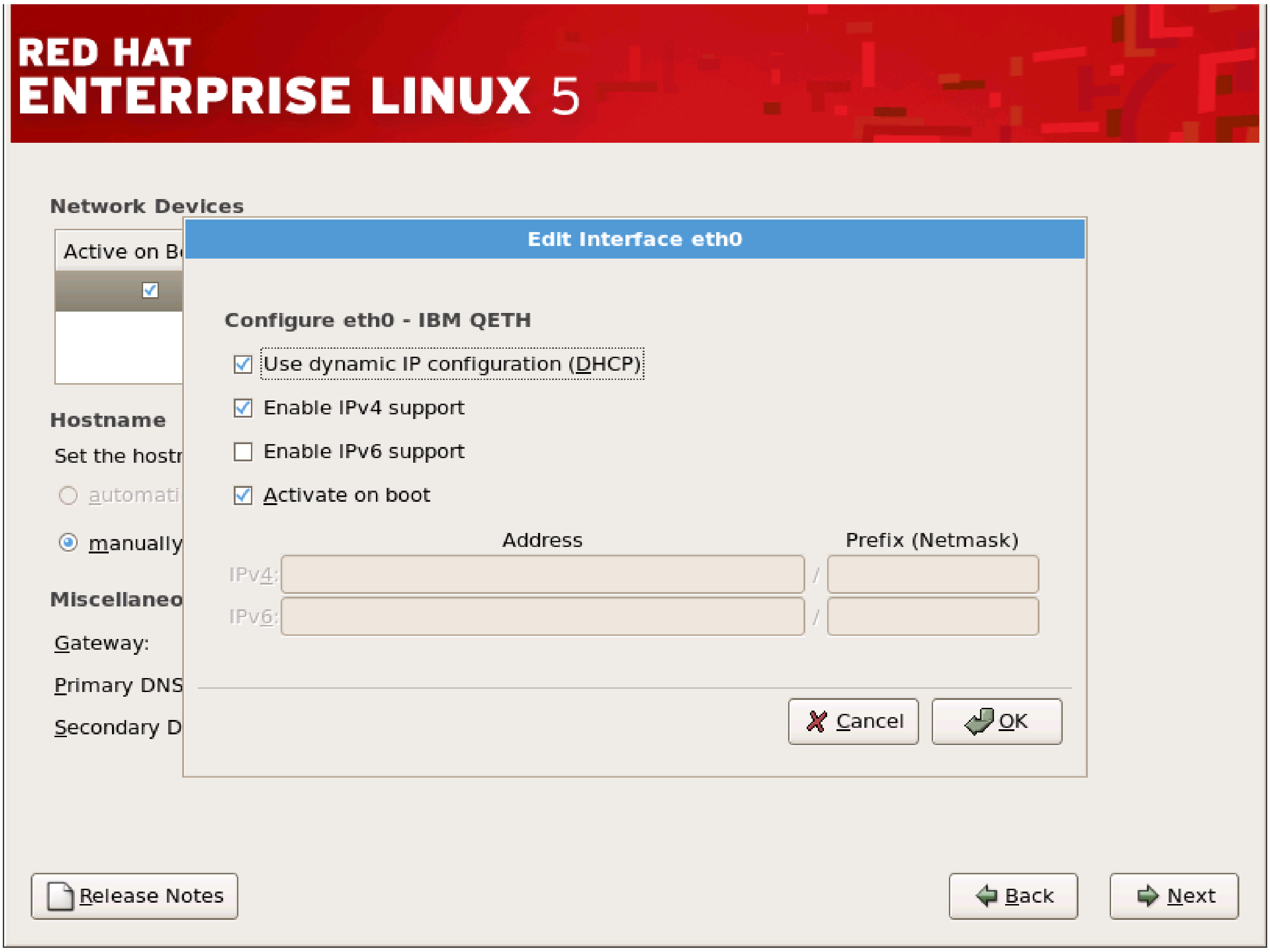

Figure 17.14. Editing a Network Device

Note

Do not use the numbers as seen in this sample configuration. These values will not work for your own network configuration. If you are not sure what values to enter, contact your network administrator for assistance.

If you have a hostname (fully qualified domain name) for the network device, you can choose to have DHCP (Dynamic Host Configuration Protocol) automatically detect it or you can manually enter the hostname in the field provided.

Finally, if you entered the IP and Netmask information manually, you may also enter the Gateway address and the Primary and Secondary DNS addresses.

17.16. Time Zone Configuration

Set your time zone by selecting the city closest to your computer's physical location. Click on the map to zoom in to a particular geographical region of the world.

From here there are two ways for you to select your time zone:

- Using your mouse, click on the interactive map to select a specific city (represented by a yellow dot). A red **X** appears indicating your selection.

- You can also scroll through the list at the bottom of the screen to select your time zone. Using your mouse, click on a location to highlight your selection.

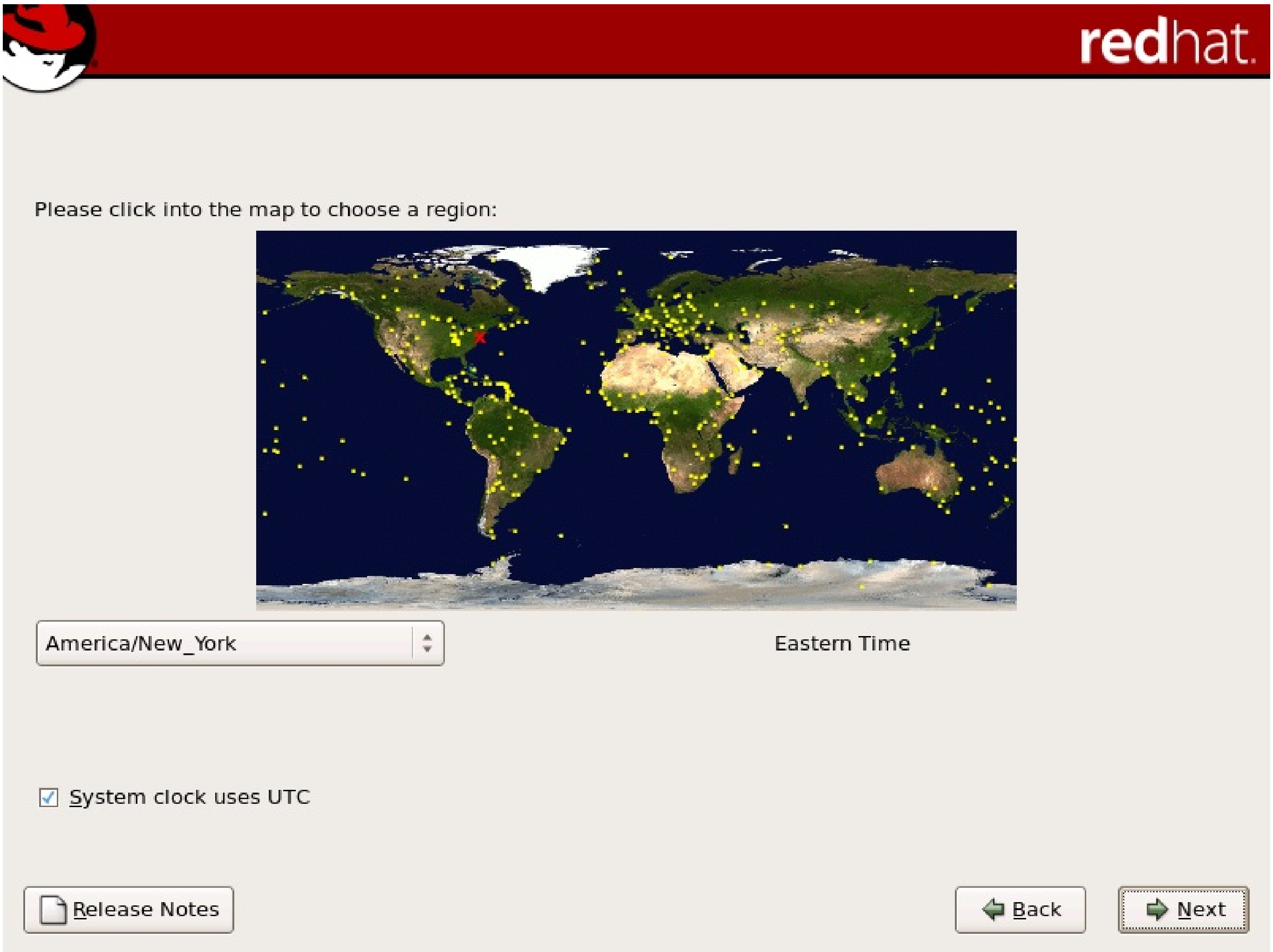

Figure 17.15. Configuring the Time Zone

Select **`System Clock uses UTC`** if you know that your system is set to UTC.

> **Note**
>
> To change your time zone configuration after you have completed the installation, use the **Time and Date Properties Tool**.
>
> Type the **`system-config-date`** command in a shell prompt to launch the **Time and Date Properties Tool**. If you are not root, it prompts you for the root password to continue.
>
> To run the **Time and Date Properties Tool** as a text-based application, use the command **`timeconfig`**.

17.17. Set Root Password

Setting up a root account and password is one of the most important steps during your installation. Your root account is similar to the administrator account used on Windows NT machines. The root account is used to install packages, upgrade RPMs, and perform most system maintenance. Logging in as root gives you complete control over your system.

Note

The root user (also known as the superuser) has complete access to the entire system; for this reason, logging in as the root user is best done *only* to perform system maintenance or administration.

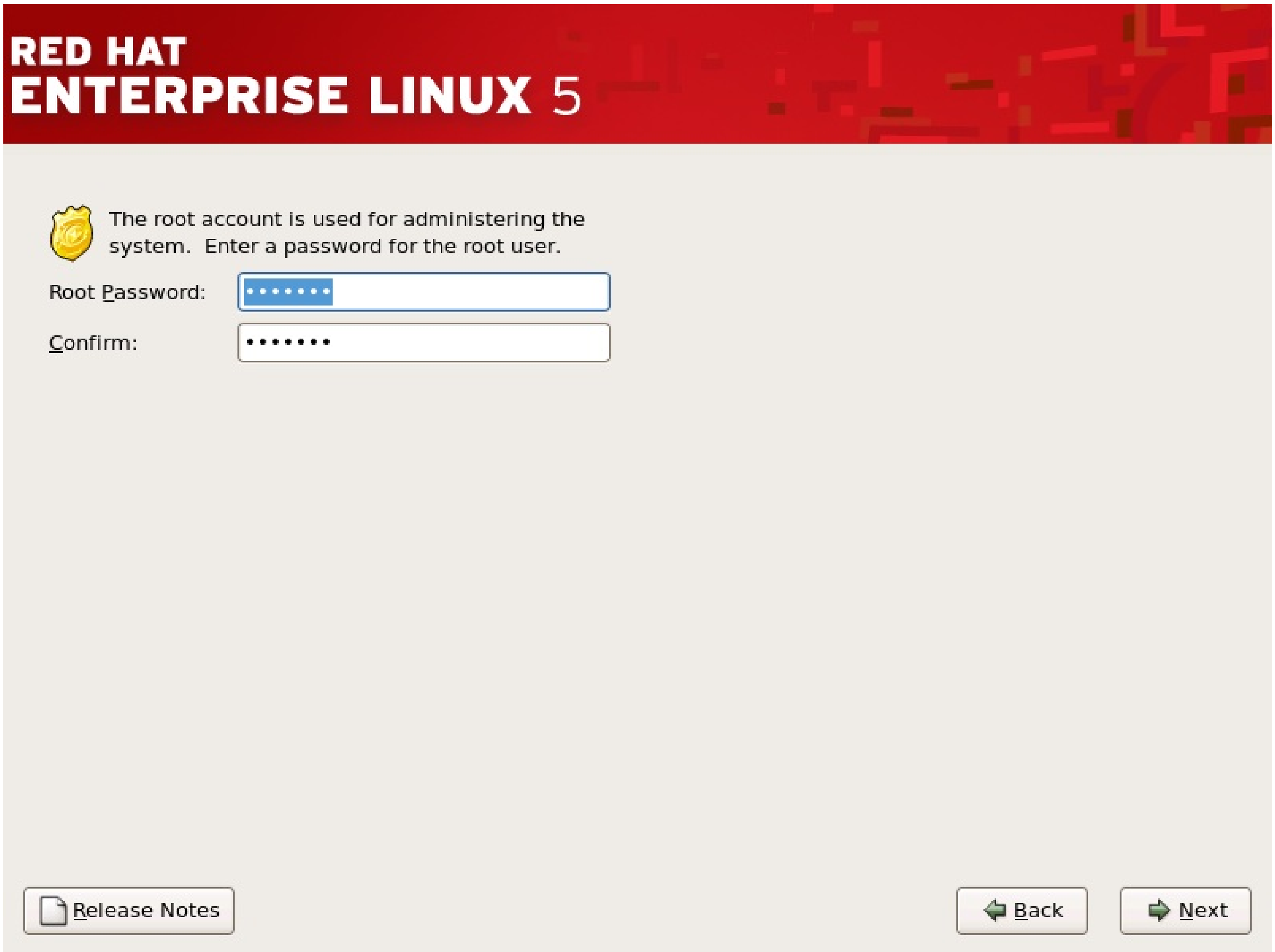

Figure 17.16. Root Password

Use the root account only for system administration. Create a non-root account for your general use and **su -** to root when you need to fix something quickly. These basic rules minimize the chances of a typo or an incorrect command doing damage to your system.

> **Note**
>
> To become root, type **su -** at the shell prompt in a terminal window and then press **Enter**. Then, enter the root password and press **Enter**.

The installation program prompts you to set a root password [10] for your system. *You cannot proceed to the next stage of the installation process without entering a root password.*

The root password must be at least six characters long; the password you type is not echoed to the screen. You must enter the password twice; if the two passwords do not match, the installation program asks you to enter them again.

You should make the root password something you can remember, but not something that is easy for someone else to guess. Your name, your phone number, *qwerty*, *password, root*, *123456*, and *anteater* are all examples of bad passwords. Good passwords mix numerals with upper and lower case letters and do not contain dictionary words: *Aard387vark* or *420BMttNT*, for example. Remember that the password is case-sensitive. If you write down your password, keep it in a secure place. However, it is recommended that you do not write down this or any password you create.

> **Note**
>
> Do not use one of the example passwords offered in this manual. Using one of these passwords could be considered a security risk.

> **Note**
>
> To change your root password after you have completed the installation, use the **Root Password Tool**.
>
> Type the **system-config-rootpassword** command in a shell prompt to launch the **Root Password Tool**. If you are not root, it prompts you for the root password to continue.

17.18. Package Group Selection

Now that you have made most of the choices for your installation, you are ready to confirm the default package selection or customize packages for your system.

The **Package Installation Defaults** screen appears and details the default package set for your Red Hat Enterprise Linux installation. This screen varies depending on the version of Red Hat Enterprise Linux you are installing.

If you choose to accept the current package list, skip ahead to Section 17.19, “Preparing to Install”.

To customize your package set further, select the **Customize now** option on the screen. Clicking **Next** takes you to the **Package Group Selection** screen.

You can select package groups, which group components together according to function (for example, **X Window System** and **Editors**), individual packages, or a combination of the two.

> **Note**
>
> Users of IBM System z who want support for developing or running legacy 31-bit applications are encouraged to select the **`Compatibility Arch Support`** and **`Compatibility Arch Development Support`** packages to install architecture specific support for their systems.

To select a component, click on the checkbox beside it (refer to Figure 17.17, "Package Group Selection").

Figure 17.17. Package Group Selection

Select each component you wish to install.

Once a package group has been selected, if optional components are available you can click on **`Optional packages`** to view which packages are installed by default, and to add or remove optional packages from that group. If there are no optional components this button will be disabled.

Figure 17.18. Package Group Details

17.19. Preparing to Install

17.19.1. Preparing to Install

A screen preparing you for the installation of Red Hat Enterprise Linux now appears.

For your reference, a complete log of your installation can be found in **`/root/install.log`** once you reboot your system.

Warning

If, for some reason, you would rather not continue with the installation process, this is your last opportunity to safely cancel the process and reboot your machine. Once you press the **`Next`** button, partitions are written and packages are installed. If you wish to abort the installation, you should close your SSH session and re-IPL the machineeboot now before any existing information on any hard drive is rewritten.

To cancel this installation process, close your SSH session and re-IPL the system using your 3270 terminal emulator.

17.20. Installing Packages

At this point there is nothing left for you to do until all the packages have been installed. How quickly this happens depends on the number of packages you have selected and your computer's speed.

17.21. Installation Complete

Congratulations! Your Red Hat Enterprise Linux installation is now complete!

The installation program prompts you to prepare your system for reboot.

Once the installation is complete, you must IPL (boot) from either the DASD or SCSI LUN where the **/boot** partition for Red Hat Enterprise Linux has been installed.

For example, using dasd at device **200** on the 3270 console you may issue the command **#cp i 200**. Often, especially in DASD only environments where automatic partitioning (clearing data from all partitions) was chosen, the first dasd (**dasda**) is where the **/boot** partition is located.

Using **/boot** on a SCSI LUN in a z/VM guest account, it may be necessary to provide WWPN and LUN information through which a zFCP device can ipl. As an example,

```
#CP SET LOADDEV PORTNAME 50050763 FCCD9689 LUN 83030000 00000000
```

could be used to provide zFCP routing information to a zFCP device (where **0x50050763FCCD9689** is the example WWPN, and **8303** is the SCSI LUN). Then the zFCP device information can be queried and used to start the IPL:

```
#cp q v fcp
```

After querying this information the zFCP device (**4322** in this example) could be ipl-ed with a command like:

```
#cp ipl 4322
```

For LPAR-based installations, the HMC console may be used to issue a load command to the LPAR, specifying the particular DASD or SCSI LUN and zFCP WWPN where the **/boot** partition is located.

> **Note**
>
> For guest accounts using z/VM, assuming you want to disconnect from the 3270 console without halting the linux guest, use **#cp disc** instead of **#cp logout** or **#cp log**. This allows for the virtual system running Red Hat Enterprise Linux for IBM System z to continue even when not connected to the 3270 console.

Following IPLing the installed Red Hat Enterprise Linux OS, you may log on to the system via **ssh**. It is important to note that the only place you can log in as root is from the 3270 or from other devices as listed in **/etc/securetty**.

The first time you start your Red Hat Enterprise Linux system in a graphical environment, the **Setup Agent** may be manually started, which guides you through the Red Hat Enterprise Linux configuration. Using this tool, you can set your system time and date, install software, register your machine with Red Hat Network, and more. The **Setup Agent** lets you configure your environment at the beginning, so that you can get started using your Red Hat Enterprise Linux system quickly.

For information on registering your Red Hat Enterprise Linux subscription, refer to [Chapter 25, *Register Your System and Apply Subscriptions*].

[10] A root password is the administrative password for your Red Hat Enterprise Linux system. You should only log in as root when needed for system maintenance. The root account does not operate within the restrictions placed on normal user accounts, so changes made as root can have implications for your entire system.

Chapter 18. Removing Red Hat Enterprise Linux

To remove from the S/390 you can either remove the DASD allocation from the VM or you can start the installation program and re-format all of the DASD partitions. Instead of selecting **OK** you will select **Cancel** to exit the installation program.

Chapter 19. Sample Parameter Files

The IBM System z architectures use a special parameter file to set up networking before the installation program (**anaconda**) can be started. This section describes the contents of the parameter file.

The parameter file has a limit of 32 total parameters. To accommodate limitations of the parameter files, a new configuration file on a CMS DASD should be used to configure the initial network setup and the DASD specification. The **.parm** file should contain the real kernel parameters, such as ***root=/dev/ram0 ro ip=off ramdisk_size=40000***, and single parameters which are not assigned to variables, such as ***vnc***. Two new parameters which point the installation program at the new configuration file need to be added to the **.parm** file. They are ***CMSDASD*** and ***CMSCONF*** .

CMSDASD=cmsdasd_address

Where *cmsdasd_address* represents the list of the device ID of the CMS DASD device which contains the configuration file. This is usually the CMS user's 'A' disk. This option is applicable only for users who have a CMS formatted disk (z/VM) available.

For example: ***CMSDASD=191***

CMSCONFFILE=configuration_file

Where *configuration_file* represents the name of the configuration file. This value must be specified in lower case. It is specified in a Linux style file name format. The CMS file **REDHAT CONF** is specified as **redhat.conf**. This option is applicable only for users who have a CMS formatted disk (z/VM) available.

For example: ***CMSCONFFILE=redhat.conf***

DASD=dasd-list

Where *dasd-list* represents the list of DASD devices to be used by Red Hat Enterprise Linux.

Although automatic probing for DASDs is done if this parameter is omitted, it is highly recommended to include the ***DASD=*** parameter, as the device numbers (and therefore the device names) can vary when a new DASD is added to the guest. This can result in an unusable system.

For example: ***DASD=0.0.0100,0.0201-0.0.0204***

The following parameters are required to set up networking:

SUBCHANNELS=

Provides required device bus IDs for the various network interfaces.

```
qeth: SUBCHANNELS="read_device_bus_id,write_device_bus_id,
      data_device_bus_id"
lcs: SUBCHANNELS="read_device_bus_id,write_device_bus_id"
```

Due to the length of the qeth command line, it has been broken into two lines.

> **Note**
>
> The CTC, and NETIUCV drivers have been deprecated and are no longer supported in Red Hat Enterprise Linux.

For example (a sample qeth SUBCHANNEL statement):

```
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
```

The following parameters are optional:

HOSTNAME=string

Where *string* is the hostname of the newly-installed Linux guest.

NETTYPE=type

Where *type* must be one of the following: qeth or lcs.

IPADDR=IP

Where *IP* is the IP address of the new Linux guest.

NETWORK=network

Where *network* is the address of your network.

NETMASK=netmask

Where *netmask* is the netmask.

BROADCAST=broadcast

Where *broadcast* is the broadcast address.

GATEWAY=gw

Where *gw* is the gateway-IP for your eth device.

MTU=mtu

Where *mtu* is the Maximum Transmission Unit (MTU) for this connection.

DNS=server1:server2:additional_server_terms:serverN

Where *server1*:*server2*:*additional_server_terms*:*serverN* is a list of DNS servers, separated by colons. For example:

```
DNS=10.0.0.1:10.0.0.2
```

SEARCHDNS=domain1:domain2:additional_dns_terms:domainN

Where *domain1*:*domain2*:*additional_dns_terms*:*domainN* is a list of the search domains, separated by colons. For example:

```
SEARCHDNS=example.com:example.org
```

PORTNAME=osa_portname | lcs_portnumber

This variable supports OSA devices operating in qdio mode or in non-qdio mode.

When using qdio mode: *osa_portname* is the portname specified on the OSA device when operating in qeth mode. PORTNAME is only required for z/VM 4.3 or older without APARs VM63308 and PQ73878.

When using non-qdio mode: *lcs_portnumber* is used to pass the relative port number as integer in the range of 0 through 15.

PORTNO=portnumber

When installing under z/VM, you can add either **PORTNO=0** (to use port 0) or **PORTNO=1** (to use port 1) to the CMS configuration file to avoid being prompted for the mode.

The ***PORTNO=*** setting also works on LPAR, but you must place it directly in the parmfile rather than the CMS configuration file.

LAYER2=

Add either **LAYER2=0** or **LAYER2=1** to the CMS configuration file to make the mode persistent when installing on a System z guest.

Use **LAYER2=0** when the OSA is in layer 3 mode, and **LAYER2=1** when the OSA is in layer 2 mode.

VSWITCH=

When specifying **LAYER2=1**, you can also specify **VSWITCH=1** when connecting to a VSWITCH, or **VSWITCH=0** when connecting directly to the OSA.

MACADDR=MAC_address

When specifying **LAYER2=1** and a VSWITCH is not in use, you can use this parameter to specify the MAC address in the CMS configuration file.

FCP_* (FCP_1, FCP_2, ...)

These variables can be used on systems with FCP devices to preconfigure the FCP setup (these can be changed during the installation).

Use the following samples as a guide to formatting proper parameter files.

Sample file with minimally required parameters:

```
root=/dev/ram0 DASD=200
```

Note

The installation program prompts the user for any required parameters not specified in the parameter file.

Sample file configuring a QETH networking device:

Example of **redhat.parm** file:

```
root=/dev/ram0 ro ip=off ramdisk_size=40000
CMSDASD=191 CMSCONFFILE=redhat.conf
vnc
```

Example of **`redhat.conf`** file (pointed to by CMSCONFFILE in **`redhat.parm`**)

```
DASD=200
HOSTNAME="foobar.systemz.example.com"
DASD="200-203"
NETTYPE="qeth"
IPADDR="192.168.17.115"
SUBCHANNELS="0.0.0600,0.0.0601,0.0.0602"
PORTNAME="FOOBAR"
NETWORK="192.168.17.0"
NETMASK="255.255.255.0"
BROADCAST="192.168.17.255"
SEARCHDNS="example.com:systemz.example.com"
GATEWAY="192.168.17.254"
DNS="192.168.17.1"
MTU="4096"
```

Chapter 20. Additional Boot Options

This section discusses additional boot and kernel boot options available for the Red Hat Enterprise Linux installation program.

To use any of the boot options presented here, type the command you wish to invoke at the installation **`boot:`** prompt.

You can also store kernel boot options for the z/IPL boot loader in the **`zipl.conf`** file, either by editing the file manually or using the **`zipl`** tool.

Kernel parameters in `zipl.conf` that contain quotation marks

Where kernel parameters use nested quotation marks, the required order in Red Hat Enterprise Linux is double quote marks for the outer set and single quote marks for the inner set. For example, **`parameters="vmhalt='LOGOFF'"`** is correct, while **`parameters='vmhalt="LOGOFF"'`** is not correct and may lead to unexpected behavior.

boot time command arguments

`askmethod`

this command asks you to select the installation method you would like to use when booting from the Red Hat Enterprise Linux CD-ROM.

`dd=`*`url`*

this argument causes the installation program to prompt you to use a driver image from a specified HTTP, FTP, or NFS network address.

`display=`*`ip`*`:0`

this command allows remote display forwarding. In this command, *ip* should be replaced with the IP address of the system on which you want the display to appear.

On the system you want the display to appear on, you must execute the command **`xhost +`*`remotehostname`***, where *remotehostname* is the name of the host from which you are running the original display. Using the command **`xhost +`*`remotehostname`*** limits access to the remote display terminal and does not allow access from anyone or any system not specifically authorized for remote access.

`mediacheck`

this command gives you the option of testing the integrity of the install source (if an ISO-based method). this command works with the CD, DVD, hard drive ISO, and NFS ISO installation methods. Verifying that the ISO images are intact before you attempt an installation helps to avoid problems that are often encountered during an installation.

`mpath`

enables multipath support.

> **Important — mandatory for installation on multipath devices**
>
> If you install Red Hat Enterprise Linux 5.11 on a network storage device accessible through multiple paths, you must boot the installation process with this option. If you do not specify this option at boot time, installation will fail, or the system will fail to boot after installation completes.

`noeject`

do not eject optical discs after installation. This option is useful in remote installations where it is difficult to close the tray afterwards.

`noprobe`

this command disables hardware detection and instead prompts the user for hardware information.

`rescue`

this command runs rescue mode. Refer to Chapter 27, *Basic System Recovery* for more information about rescue mode.

`text`

this command disables the graphical installation program and forces the installation program to run in text mode.

`vnc`

this command allows you to install from a VNC server.

`vncpassword=`

this command sets the password used to connect to the VNC server.

`noipv6`

this command disables the default selection of ipv6 options during the installer stage 1 processing. Ipv6 settings may still be made manually if this option is specified, but the default behavior will be that Ipv6 settings are not enabled.

`cmdline`

The 3270 console (most often used during installation on IBM System z) does not recognize terminal formatting entries common to most unix-style terminals. Specifying this option changes the behavior of anaconda during kickstart installations so that console output on the 3270 is much better. This option should not be used for regular, interactive installations.

`RUNKS=1`

This option is used to specify (usually in conjunction with the **`cmdline`** option) kickstart installation for IBM System z.

Chapter 21. Troubleshooting Installation on an IBM System z System

This appendix discusses some common installation problems and their solutions.

21.1. You are Unable to Boot Red Hat Enterprise Linux

21.1.1. Is Your System Displaying Signal 11 Errors?

A signal 11 error, commonly know as a *segmentation fault*, means that the program accessed a memory location that was not assigned to it. A signal 11 error may be due to a bug in one of the software programs that is installed, or faulty hardware.

Ensure that you have the latest installation updates and images from Red Hat. Review the online errata to see if newer versions are available.

21.2. Trouble During the Installation

21.2.1. `No devices found to install Red Hat Enterprise Linux` Error Message

If you receive an error message stating **`No devices found to install Red Hat Enterprise Linux`**, then there may be an issue with your DASD devices. If you encounter this error, add the **`DASD=<disks>`** parameter to your **`parm`** file (where *disks* is the DASD range reserved for installation) and start the install again.

Additionally, make sure you format the DASDs using the **`dasdfmt`** command within a Linux root shell, instead of formatting the DASDs using CMS.

21.2.2. Trouble with Partition Tables

If you receive an error after the **`Disk Partitioning Setup`** (Section 17.11, “Disk Partitioning Setup”) phase of the installation saying something similar to

> The partition table on device hda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive.

you may not have a partition table on that drive or the partition table on the drive may not be recognizable by the partitioning software used in the installation program.

No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

21.2.3. Other Partitioning Problems

If you are using **Disk Druid** to create partitions, but cannot move to the next screen, you probably have not created all the partitions necessary for **Disk Druid**'s dependencies to be satisfied.

You must have the following partitions as a bare minimum:

- A / (root) partition

- A <swap> partition of type swap

> **Note**
>
> When defining a partition's type as swap, do not assign it a mount point. **Disk Druid** automatically assigns the mount point for you.

21.2.4. Are You Seeing Python Errors?

During some upgrades or installations of Red Hat Enterprise Linux, the installation program (also known as **anaconda**) may fail with a Python or traceback error. This error may occur after the selection of individual packages or while trying to save the upgrade log in the **/tmp/**directory. The error may look similar to:

```
Traceback (innermost last):
File "/var/tmp/anaconda-7.1//usr/lib/anaconda/iw/progress_gui.py", line 20, in run
rc = self.todo.doInstall ()
File "/var/tmp/anaconda-7.1//usr/lib/anaconda/todo.py", line 1468, in doInstall
self.fstab.savePartitions ()
File "fstab.py", line 221, in savePartitions
sys.exit(0)
SystemExit: 0
Local variables in innermost frame:
self: <fstab.GuiFstab instance at 8446fe0>
sys: <module 'sys' (built-in)>
ToDo object:  (itodo  ToDo  p1  (dp2  S'method'  p3  (iimage  CdromInstallMethod
p4  (dp5  S'progressWindow'  p6   <failed>
```

This error occurs in some systems where links to **/tmp/** are symbolic to other locations or have been changed since creation. These symbolic or changed links are invalid during the installation process, so the installation program cannot write information and fails.

If you experience such an error, first try to download any available errata for **anaconda**. Errata can be found at:

```
http://www.redhat.com/support/errata/
```

The **anaconda** website may also be a useful reference and can be found online at:

```
http://fedoraproject.org/wiki/Anaconda
```

You can also search for bug reports related to this problem. To search Red Hat's bug tracking system, go to:

```
http://bugzilla.redhat.com/bugzilla/
```

Finally, if you are still facing problems related to this error, register your product and contact our support team. To register your product, go to:

```
http://www.redhat.com/apps/activate/
```

21.3. Problems After Installation

21.3.1. Remote Graphical Desktops and XDMCP

If you have installed the X Window System and would like to log in to your Red Hat Enterprise Linux system using a graphical login manager, enable the *X Display Manager Control Protocol* (XDMCP). This protocol allows users to remotely log in to a desktop environment from any X Window System compatible client (such as a network-connected workstation or X terminal). To enable remote login using XDMCP, edit the following line in the **/etc/gdm/custom.conf** file on the Red Hat Enterprise Linux system with a text editor such as **vi** or **nano**:

Add the line **Enable=true**, save the file, and exit the text editor. Switch to runlevel 5 to start the X server:

```
/sbin/init 5
```

From the client machine, start remote X session using **X**. For example:

```
X :1 -query s390vm.example.com
```

The command connects to the remote X server via XDMCP (replace *s390vm.example.com* with the hostname of the remote X server) and displays the remote graphical login screen on display **:1** of the client system (usually accessible by using the **Ctrl**-**Alt**-**F8** key combination).

You may also access remote desktop sessions using a *nested* X server, which opens the remote desktop as a window in your current X session. **Xnest** allows users to open a remote desktop nested within their local X session. For example, run **Xnest** using the following command, replacing *s390vm.example.com* with the hostname of the remote X server:

```
Xnest :1 -query s390vm.example.com
```

21.3.2. Problems When You Try to Log In

If you did not create a user account in the **Setup Agent**, log in as root and use the password you assigned to root.

If you cannot remember your root password, boot your system as **linux single**.

Once you have booted into single user mode and have access to the **#** prompt, you must type **passwd root**, which allows you to enter a new password for root. At this point you can type **shutdown -r now** to reboot the system with the new root password.

If you cannot remember your user account password, you must become root. To become root, type **su -** and enter your root password when prompted. Then, type **passwd <username>**. This allows you to enter a new password for the specified user account.

If the graphical login screen does not appear, check your hardware for compatibility issues. The *Hardware Compatibility List* can be found at:

```
http://hardware.redhat.com/hcl/
```

21.3.3. Your Printer Does Not Work

If you are not sure how to set up your printer or are having trouble getting it to work properly, try using the **Printer Configuration Tool**.

Type the **`system-config-printer`** command at a shell prompt to launch the **Printer Configuration Tool**. If you are not root, it prompts you for the root password to continue.

21.3.4. Apache-based `httpd` service/Sendmail Hangs During Startup

If you are having trouble with the Apache-based **`httpd`** service or Sendmail hanging at startup, make sure the following line is in the **`/etc/hosts`** file:

```
127.0.0.1  localhost.localdomain  localhost
```

Chapter 22. Additional Information for IBM System z Users

22.1. The sysfs File System

The Linux 2.6 kernel introduced the **sysfs** file system. The **sysfs** file system is described as a union of the **proc**, **devfs**, and **devpty** file systems. The **sysfs** file system enumerates the devices and busses attached to the system into a file system hierarchy that can be accessed from user space. It is designed to handle the device and driver specific options that have previously resided in **/proc/**, and encompass the dynamic device addition previously offered by **devfs**.

The **sysfs** file system is mounted at **/sys/** and contains directories that organize the devices attached to the system in several different ways. The **/sysfs/** subdirectories include:

1. The **/devices/** directory

   This directory contains the **/css0/** directory. Its subdirectories represent all the subchannels detected by the Linux kernel. Subchannel directories are named in the form **0.0.*nnnn*** where *nnnn* is the subchannel number in hex between 0000 and ffff. Subchannel directories in turn contain status files and another subdirectory which represents the actual device. The device directory is named **0.0.*xxxx*** where *xxxx* is the unit address for the device. The **/devices/** directory also contains status information as well as configuration options for the device.

2. The **/bus/** directory

   This contains a **/ccw/** subdirectory and a **/ccwgroup/** subdirectory. CCW devices are accessed using channel command words. Devices in the **/ccw/** directory only use one subchannel on the mainframe channel subsystem. CCW group devices are also accessed with channel command words, but they use more than one subchannel per device. For example, a 3390-3 DASD device uses one subchannel, while a QDIO network connection for an OSA adapter uses three subchannels. The **/ccw/** and the **/ccwgroup/** directories both contain directories called devices and drivers:

   The **/devices/** directory contains a symbolic link to the device directories in the **/sys/devices/css0/** directory.

   The **/drivers/** directory contains directories for each device driver currently loaded on the system. Drivers associated with devices such as **dasd**, **console**, **qeth**, and **zfcp** have directory entries here. The **/driver/** directory contains settings for the device driver, as well as symbolic links to the devices it is using (in the **/sys/devices/css0/** directory).

3. The **/class/** directory

   This contains directories that group together similar devices such as ttys, SCSI tape drives, network devices, and other miscellaneous devices.

4. The **/block/** directory

   This directory contains directories for each of the block devices on the system. These are mostly disk type devices such as real DASD, loopback devices, and software raid block devices. The noticeable difference between older Linux systems and ones that use **sysfs** is the need to refer to devices by their **sysfs** name. On a 2.4 kernel image, the **zFCP** driver was passed as its device addresses. On the 2.6 Kernel image system the driver is passed as **0.0.1600**.

22.2. Using the zFCP Driver

During the initial installation, you are prompted to enter SCSI/FCP information. If this information is entered, it creates the **/etc/zfcp.conf** file which contains your SCSI configuration. It also adds the line **alias scsi_hostadapter zFCP** to **/etc/modprobe.conf**. This loads the required **zFCP** modules.

```
# cat /etc/zfcp.conf
0.0.010a 0x01 0x5005076300c18154 0x00 0x5719000000000000

# cat /etc/modprobe.conf
alias eth0 qeth
options dasd_mod dasd=201,4b2e
alias scsi_hostadapter zfcp
```

If no SCSI devices were defined during the initial installation, the following example demonstrates how to add one manually:

```
# cd /lib/modules/2.6.7-1.451.2.3/kernel/drivers/s390/scsi
# modprobe zfcp

# lsmod
Module                    Size  Used by
zfcp                    221460  0 [permanent]
autofs4                  39944  0
qeth                    166288  0
qdio                     60240  3 zfcp,qeth
ccwgroup                 25344  1 qeth
ipt_REJECT               23552  1
ipt_state                18944  5
ip_conntrack             57904  1 ipt_state
iptable_filter           19712  1
ip_tables                37888  3 ipt_REJECT,ipt_state,iptable_filter
sd_mod                   39688  0
scsi_mod                182904  2 zfcp,sd_mod
dm_mod                   86408  0
ext3                    179056  2
jbd                      92720  1 ext3
dasd_fba_mod             25344  0
dasd_eckd_mod            77056  4
dasd_mod                 85328  6 dasd_fba_mod,dasd_eckd_mod

# cd /sys/bus/ccw/drivers/zfcp/0.0.010a

# echo 1 > online
# cat online
1

# echo 0x5005076300c18154 > /sys/bus/ccw/drivers/zfcp/0.0.010a/port_add
# ls
0x5005076300c18154  failed           lic_version    s_id
availability        fc_link_speed    nameserver     status
card_version        fc_service_class online         wwnn
cmb_enable          fc_topology      port_add       wwpn
cutype              hardware_version port_remove
detach_state        host2            scsi_host_no
devtype             in_recovery      serial_number

# cd /sys/bus/ccw/drivers/zfcp/0.0.010a/0x5005076300c18154
# echo 0x5719000000000000 > unit_add
# ls
0x5719000000000000  d_id    in_recovery  status    unit_remove
detach_state        failed  scsi_id      unit_add  wwnn
```

```
# cat /sys/bus/ccw/drivers/zfcp/0.0.010a/scsi_host_no
0x0
# cat /sys/bus/ccw/drivers/zfcp/0.0.010a/0x5005076300c18154/scsi_id
0x1
# cat \
/sys/bus/ccw/drivers/zfcp/0.0.010a/0x5005076300c18154/0x5719000000000000/scsi_lun
0x0

# cat /sys/bus/scsi/devices/0\:0\:1\:0/hba_id
0.0.010a
# cat /sys/bus/scsi/devices/0\:0\:1\:0/wwpn
0x5005076300c18154
# cat /sys/bus/scsi/devices/0\:0\:1\:0/fcp_lun
0x5719000000000000

# cat /sys/bus/scsi/devices/0\:0\:1\:0/block/dev
8:0
# cat /sys/bus/scsi/devices/0\:0\:1\:0/block/sda1/dev
8:1

# cat /proc/scsi/scsi
Attached devices:
Host: scsi2 Channel: 00 Id: 01 Lun: 00
  Vendor: IBM      Model: 2105F20          Rev: .123
  Type:   Direct-Access                    ANSI SCSI revision: 03

# fdisk /dev/sda

# mke2fs -j /dev/sda1

# mount /dev/sda1 /mnt
# df
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/dasda1            2344224   1427948    797196  65% /
none                    511652         0    511652   0% /dev/shm
/dev/dasdb1            2365444     32828   2212456   2% /opt
/dev/sda1              3844088     32828   3615988   1% /mnt

# cd /boot
# mv initrd-2.6.7-1.451.2.3.img initrd-2.6.7-1.451.2.3.img.orig
# mkinitrd -v --with=scsi_mod --with=zfcp --with=sd_mod initrd-2.6.7-1.451.2.3.img
2.6.7-1.451.2.3
Looking for deps of module ide-disk
Looking for deps of module dasd_mod
Looking for deps of module dasd_eckd_mod         dasd_mod
Looking for deps of module dasd_mod
Looking for deps of module dasd_fba_mod  dasd_mod
Looking for deps of module dasd_mod
Looking for deps of module ext3  jbd
Looking for deps of module jbd
Looking for deps of module scsi_mod
Looking for deps of module zfcp  qdio scsi_mod
Looking for deps of module qdio
Looking for deps of module scsi_mod
Looking for deps of module sd_mod        scsi_mod
Looking for deps of module scsi_mod
Using modules:  ./kernel/drivers/s390/block/dasd_mod.ko
./kernel/drivers/s390/block/dasd_eckd_mod.ko
./kernel/drivers/s390/block/dasd_fba_mod.ko ./kernel/fs/jbd/jbd.ko
./kernel/fs/ext3/ext3.ko ./kernel/drivers/scsi/scsi_mod.ko
./kernel/drivers/s390/cio/qdio.ko ./kernel/drivers/s390/scsi/zfcp.ko
./kernel/drivers/scsi/sd_mod.ko
Using loopback device /dev/loop0
/sbin/nash -> /tmp/initrd.cT1534/bin/nash
/sbin/insmod.static -> /tmp/initrd.cT1534/bin/insmod
`/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/s390/block/dasd_mod.ko'->
```

```
`/tmp/initrd.cT1534/lib/dasd_mod.ko'
`/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/s390/block/dasd_eckd_mod.ko' ->
`/tmp/initrd.cT1534/lib/dasd_eckd_mod.ko'
`/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/s390/block/dasd_fba_mod.ko' ->
`/tmp/initrd.cT1534/lib/dasd_fba_mod.ko'
`/lib/modules/2.6.7-1.451.2.3/./kernel/fs/jbd/jbd.ko' ->
`/tmp/initrd.cT1534/lib/jbd.ko'
`/lib/modules/2.6.7-1.451.2.3/./kernel/fs/ext3/ext3.ko' ->
`/tmp/initrd.cT1534/lib/ext3.ko'
`/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/scsi/scsi_mod.ko' ->
`/tmp/initrd.cT1534/lib/scsi_mod.ko'
`/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/s390/cio/qdio.ko' ->
`/tmp/initrd.cT1534/lib/qdio.ko'
`/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/s390/scsi/zfcp.ko' ->
`/tmp/initrd.cT1534/lib/zfcp.ko'
`/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/scsi/sd_mod.ko' ->
`/tmp/initrd.cT1534/lib/sd_mod.ko'
...
Loading module dasd_mod with options dasd=201,4b2e
Loading module dasd_eckd_mod
Loading module dasd_fba_mod
Loading module jbd
Loading module ext3
Loading module scsi_mod
Loading module qdio
Loading module zfcp
Loading module sd_mod

# zipl -V
Using config file '/etc/zipl.conf'
Target device information
  Device..........................: 5e:00
  Partition.......................: 5e:01
  Device name.....................: dasda
  DASD device number..............: 0201
  Type............................: disk partition
  Disk layout.....................: ECKD/compatible disk layout
  Geometry - heads................: 15
  Geometry - sectors..............: 12
  Geometry - cylinders............: 3308
  Geometry - start................: 24
  File system block size..........: 4096
  Physical block size.............: 4096
  Device size in physical blocks..: 595416
Building bootmap '/boot//bootmap'
Building menu 'rh-automatic-menu'
Adding #1: IPL section 'linux' (default)
  kernel image......: /boot/vmlinuz-2.6.7-1.451.2.3 at 0x10000
  kernel parmline...: 'root=LABEL=/' at 0x1000
  initial ramdisk...: /boot/initrd-2.6.7-1.451.2.3.img at 0x800000
Preparing boot device: dasda (0201).
Preparing boot menu
  Interactive prompt......: disabled
  Menu timeout............: disabled
  Default configuration...: 'linux'
Syncing disks...
Done.
```

22.3. Using `mdadm` to Configure RAID-Based and Multipath Storage

Similar to other tools comprising the **`raidtools`** package set, the **`mdadm`** command can be used to perform all the necessary functions related to administering multiple-device sets. This section explains how **`mdadm`** can be used to:

- Create a RAID device
- Create a multipath device

22.3.1. Creating a RAID Device With mdadm

To create a RAID device, edit the **/etc/mdadm.conf** file to define appropriate ***DEVICE*** and ***ARRAY*** values:

```
DEVICE /dev/sd[abcd]1
ARRAY /dev/md0 devices=/dev/sda1,/dev/sdb1,/dev/sdc1,/dev/sdd1
```

In this example, the ***DEVICE*** line is using traditional file name globbing (refer to the **glob**(7) man page for more information) to define the following SCSI devices:

- **/dev/sda1**
- **/dev/sdb1**
- **/dev/sdc1**
- **/dev/sdd1**

The ***ARRAY*** line defines a RAID device (**/dev/md0**) that is comprised of the SCSI devices defined by the ***DEVICE*** line.

Prior to the creation or usage of any RAID devices, the **/proc/mdstat** file shows no active RAID devices:

```
Personalities :
read_ahead not set
Event: 0
unused devices: none
```

Next, use the above configuration and the **mdadm** command to create a RAID 0 array:

```
mdadm -C /dev/md0 --level=raid0 --raid-devices=4 /dev/sda1 /dev/sdb1 /dev/sdc1 \
/dev/sdd1
Continue creating array? yes
mdadm: array /dev/md0 started.
```

Once created, the RAID device can be queried at any time to provide status information. The following example shows the output from the command **mdadm --detail /dev/md0**:

```
/dev/md0:
Version : 00.90.00
Creation Time : Mon Mar  1 13:49:10 2004
Raid Level : raid0
Array Size : 15621632 (14.90 GiB 15.100 GB)
Raid Devices : 4
Total Devices : 4
Preferred Minor : 0
Persistence : Superblock is persistent

Update Time : Mon Mar  1 13:49:10 2004
State : dirty, no-errors
Active Devices : 4
Working Devices : 4
Failed Devices : 0
```

```
Spare Devices : 0

Chunk Size : 64K

      Number   Major   Minor   RaidDevice State
         0       8        1        0      active sync   /dev/sda1
         1       8       17        1      active sync   /dev/sdb1
         2       8       33        2      active sync   /dev/sdc1
         3       8       49        3      active sync   /dev/sdd1
           UUID : 25c0f2a1:e882dfc0:c0fe135e:6940d932
         Events : 0.1
```

22.3.2. Creating a Multipath Device With mdadm

In addition to creating RAID arrays, **mdadm** can also be used to take advantage of hardware supporting more than one I/O path to individual SCSI LUNs (disk drives). The goal of multipath storage is continued data availability in the event of hardware failure or individual path saturation. Because this configuration contains multiple paths (each acting as an independent virtual controller) accessing a common SCSI LUN (disk drive), the Linux kernel detects each shared drive once "through" each path. In other words, the SCSI LUN (disk drive) known as **/dev/sda** may also be accessible as **/dev/sdb**, **/dev/sdc**, and so on, depending on the specific configuration.

To provide a single device that can remain accessible if an I/O path fails or becomes saturated, **mdadm** includes an additional parameter to its **level** option. This parameter ***multipath*** directs the md layer in the Linux kernel to re-route I/O requests from one pathway to another in the event of an I/O path failure.

To create a multipath device, edit the **/etc/mdadm.conf** file to define values for the ***DEVICE*** and ***ARRAY*** lines that reflect your hardware configuration.

> **Note**
>
> Unlike the previous RAID example (where each device specified in **/etc/mdadm.conf** must represent different physical disk drives), each device in this file refers to the same shared disk drive.

The command used for the creation of a multipath device is similar to that used to create a RAID device; the difference is the replacement of a RAID level parameter with the ***multipath*** parameter:

```
mdadm -C /dev/md0 --level=multipath --raid-devices=4 /dev/sda1 /dev/sdb1
 /dev/sdc1 /dev/sdd1
Continue creating array? yes
mdadm: array /dev/md0 started.
```

Due to the length of the mdadm command line, it has been broken into two lines.

In this example, the hardware consists of one SCSI LUN presented as four separate SCSI devices, each accessing the same storage by a different pathway. Once the multipath device **/dev/md0** is created, all I/O operations referencing **/dev/md0** are directed to **/dev/sda1**, **/dev/sdb1**, **/dev/sdc1**, or **/dev/sdd1** (depending on which path is currently active and operational).

The configuration of **/dev/md0** can be examined more closely using the command **mdadm --detail /dev/md0** to verify that it is, in fact, a multipath device:

```
/dev/md0:
```

```
Version : 00.90.00
Creation Time : Tue Mar  2 10:56:37 2004
Raid Level : multipath
Array Size : 3905408 (3.72 GiB 3.100 GB)
Raid Devices : 1
Total Devices : 4
Preferred Minor : 0
Persistence : Superblock is persistent

Update Time : Tue Mar  2 10:56:37 2004
State : dirty, no-errors
Active Devices : 1
Working Devices : 4
Failed Devices : 0
Spare Devices : 3

    Number   Major   Minor   RaidDevice State
       0       8       49        0      active sync   /dev/sdd1
       1       8       17        1      spare   /dev/sdb1
       2       8       33        2      spare   /dev/sdc1
       3       8        1        3      spare   /dev/sda1
           UUID : 4b564608:fa01c716:550bd8ff:735d92dc
         Events : 0.1
```

Another feature of **mdadm** is the ability to force a device (be it a member of a RAID array or a path in a multipath configuration) to be removed from an operating configuration. In the following example, **/dev/sda1** is flagged as being faulty, is then removed, and finally is added back into the configuration. For a multipath configuration, these actions would not affect any I/O activity taking place at the time:

```
# mdadm /dev/md0 -f /dev/sda1
mdadm: set /dev/sda1 faulty in /dev/md0
# mdadm /dev/md0 -r /dev/sda1
mdadm: hot removed /dev/sda1
# mdadm /dev/md0 -a /dev/sda1
mdadm: hot added /dev/sda1
#
```

22.4. Configuring IPL from a SCSI Device

Anaconda (the installation program) supports the direct installation to SCSI devices. This section includes information on how to IPL from a SCSI device within z/VM.

22.4.1. IPL the SCSI Disk

To IPL the SCSI disk, we provide the WWPN and LUN to the machine loader using the **SET LOADDEV** command.

```
#cp set loaddev portname 50050763 00c18154 lun 57190000 00000000
Ready; T=0.01/0.01 15:47:53
q loaddev
PORTNAME 50050763 00C18154    LUN  57190000 00000000    BOOTPROG 0
BR_LBA   00000000 00000000
Ready; T=0.01/0.01 15:47:56
```

IPL the SCSI disk using the FCP device defined to the guest.

```
q fcp
00: FCP  010A ON FCP   010ACHPID C1 SUBCHANNEL = 0000
```

```
00:      010A QDIO-ELIGIBLE       QIOASSIST-ELIGIBLE
Ready; T=0.01/0.01 15:51:29

i 010a
00: I 010A
00: HCPLDI2816I Acquiring the machine loader from the processor
controller.
00: HCPLDI2817I Load completed from the processor controller.
00: HCPLDI2817I Now starting machine loader version 0001.
01: HCPGSP2630I The virtual machine is placed in CP mode due to a SIGP
stop and
store status from CPU 00.
00: MLOEVL012I: Machine loader up and running (version 0.13).
00: MLOPDM003I: Machine loader finished, moving data to final storage
location.
Linux version 2.6.7-1.451.2.3 (bhcompile@example.z900.redhat.com) (gcc
version 3.4
.1 20040702 (Red Hat Linux 3.4.1-2)) #1 SMP Wed Jul 14 17:52:22 EDT 2004
We are running under VM (64 bit mode)
```

Note

The example may vary slightly from your installed system due to the code available during the documentation process for this manual.

22.5. Adding DASD

The following is an example of how to add a DASD volume:

Note

Make sure the device is attached or linked to the Linux system if running under VM.

```
CP LINK RHEL4X 4B2E 4B2E MR
DASD 4B2E LINKED R/W
```

Procedure 22.1. Bringing a disk online

1. Use the **cd** command to change to the **/sys/** directory that represents that volume:

   ```
   # cd /sys/bus/ccw/drivers/dasd-eckd/0.0.4b2e/
   # ls -l
   total 0
   -r--r--r--  1 root root 4096 Aug 25 17:04 availability
   -rw-r--r--  1 root root 4096 Aug 25 17:04 cmb_enable
   -r--r--r--  1 root root 4096 Aug 25 17:04 cutype
   -rw-r--r--  1 root root 4096 Aug 25 17:04 detach_state
   -r--r--r--  1 root root 4096 Aug 25 17:04 devtype
   -r--r--r--  1 root root 4096 Aug 25 17:04 discipline
   -rw-r--r--  1 root root 4096 Aug 25 17:04 online
   -rw-r--r--  1 root root 4096 Aug 25 17:04 readonly
   -rw-r--r--  1 root root 4096 Aug 25 17:04 use_diag
   ```

2. Next, check to see if it is already online:

```
# cat online
0
```

3. If it is not online, run the following command to bring it online:

```
# echo 1 > online
# cat online
1
```

4. Verify which block devnode it is being accessed as:

```
# ls -l
total 0
-r--r--r--  1 root root 4096 Aug 25 17:04 availability
lrwxrwxrwx  1 root root    0 Aug 25 17:07 block -> ../../../../block/dasdb
-rw-r--r--  1 root root 4096 Aug 25 17:04 cmb_enable
-r--r--r--  1 root root 4096 Aug 25 17:04 cutype
-rw-r--r--  1 root root 4096 Aug 25 17:04 detach_state
-r--r--r--  1 root root 4096 Aug 25 17:04 devtype
-r--r--r--  1 root root 4096 Aug 25 17:04 discipline
-rw-r--r--  1 root root    0 Aug 25 17:04 online
-rw-r--r--  1 root root 4096 Aug 25 17:04 readonly
-rw-r--r--  1 root root 4096 Aug 25 17:04 use_diag
```

As shown in this example, device 4B2E is being accessed as **/dev/dasdb**.

As an alternative, the recommended method for bringing a disk online (automatically) is to use the following simple command:

```
# chccwdev -e 4b2e
```

Once the disk is online, change back to the **/root** directory and format the device:

```
# cd
# dasdfmt -b 4096 -d cdl -f /dev/dasdb -l LX4B2E -p -y

cyl    97 of  3338 |#----------------------------------------------|   2%
```

When the progress bar reaches the end and the format is complete, use **fdasd** to partition the device:

```
# fdasd -a /dev/dasdb
auto-creating one partition for the whole disk...
writing volume label...
writing VTOC...
checking !
wrote NATIVE!
rereading partition table...
```

Next, make a file system on the new partition:

```
# mke2fs -j /dev/dasdb1
mke2fs 1.35 (28-Feb-2004)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
300960 inodes, 600816 blocks
30040 blocks (5.00%) reserved for the super user
First data block=0
19 block groups
```

```
32768 blocks per group, 32768 fragments per group
15840 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 39 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

Mount the new file system:

```
# mount /dev/dasdb1 /opt
# mount
/dev/dasda1 on / type ext3 (rw)
none on /proc type proc (rw)
none on /sys type sysfs (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
none on /dev/shm type tmpfs (rw)
/dev/dasdb1 on /opt type ext3 (rw)
```

Add an entry to **/etc/fstab** so that the file system is mounted at IPL time:

```
# vi /etc/fstab
# cat /etc/fstab
LABEL=/                 /                       ext3    defaults
1 1
none                    /dev/pts                devpts  gid=5,mode=620
0 0
none                    /dev/shm                tmpfs   defaults
0 0
none                    /proc                   proc    defaults
0 0
none                    /sys                    sysfs   defaults
0 0
/dev/dasdb1             /opt                    ext3    defaults
1 2
```

Add the device to the option line for the dasd_mod in **/etc/modprobe.conf** Make sure to add the new device at the end of the list, otherwise it changes the *device number : devnode* mapping and file systems are not on the devices they used to be on.

```
# vi /etc/modprobe.conf
# cat /etc/modprobe.conf
alias eth0 qeth
options dasd_mod dasd=201,4B2E
```

Rerun **mkinitrd** to pick up the changes to **modprobe.conf** so that the device can be online and mountable after the next IPL:

Note that the example below has been modified slightly for readability and for printing purposes. Each line that ends with "(elf64-s390)" should be treated as one line with no spaces, such as **/tmp/initrd.AR1182/lib/dasd_mod.ko(elf64-s390)**.

```
# cd /boot
# mv initrd-2.6.7-1.451.2.3.img initrd-2.6.7-1.451.2.3.img.old
# mkinitrd -v initrd-2.6.7-1.451.2.3.img 2.6.7-1.451.2.3
Looking for deps of module ide-disk
Looking for deps of module dasd_mod
```

```
Looking for deps of module dasd_eckd_mod         dasd_mod
Looking for deps of module dasd_mod
Looking for deps of module dasd_fba_mod  dasd_mod
Looking for deps of module dasd_mod
Looking for deps of module ext3  jbd
Looking for deps of module jbd
Using modules:  ./kernel/drivers/s390/block/dasd_mod.ko
./kernel/drivers/s390/block/dasd_eckd_mod.ko
./kernel/drivers/s390/block/dasd_fba_mod.ko ./kernel/fs/jbd/jbd.ko
./kernel/fs/ext3/ext3.ko
Using loopback device /dev/loop0
/sbin/nash -> /tmp/initrd.AR1182/bin/nash
/sbin/insmod.static -> /tmp/initrd.AR1182/bin/insmod
copy from
/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/s390/block/dasd_mod.ko
(elf64-s390) to
/tmp/initrd.AR1182/lib/dasd_mod.ko(elf64-s390)
copy from
/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/s390/block/dasd_eckd_mod.ko
(elf64-s390) to
/tmp/initrd.AR1182/lib/dasd_eckd_mod.ko
(elf64-s390)
copy from
/lib/modules/2.6.7-1.451.2.3/./kernel/drivers/s390/block/dasd_fba_mod.ko
(elf64-s390) to
/tmp/initrd.AR1182/lib/dasd_fba_mod.ko
(elf64-s390)
copy from
/lib/modules/2.6.7-1.451.2.3/./kernel/fs/jbd/jbd.ko(elf64-s390) to
/tmp/initrd.AR1182/lib/jbd.ko(elf64-s390)
copy from
/lib/modules/2.6.7-1.451.2.3/./kernel/fs/ext3/ext3.ko(elf64-s390) to
/tmp/initrd.AR1182/lib/ext3.ko(elf64-s390)
Loading module dasd_mod with options dasd=201,4B2E
Loading module dasd_eckd_mod
Loading module dasd_fba_mod
Loading module jbd
Loading module ext3
```

Run **zipl** to save the changes to **initrd** for the next IPL:

```
# zipl -V
Using config file '/etc/zipl.conf'
Target device information
  Device..........................: 5e:00
  Partition.......................: 5e:01
  Device name.....................: dasda
  DASD device number..............: 0201
  Type............................: disk partition
  Disk layout.....................: ECKD/compatible disk layout
  Geometry - heads................: 15
  Geometry - sectors..............: 12
  Geometry - cylinders............: 3308
  Geometry - start................: 24
  File system block size..........: 4096
  Physical block size.............: 4096
  Device size in physical blocks..: 595416
Building bootmap '/boot//bootmap'
Building menu 'rh-automatic-menu'
Adding #1: IPL section 'linux' (default)
  kernel image......: /boot/vmlinuz-2.6.7-1.451.2.3 at 0x10000
  kernel parmline...: 'root=LABEL=/' at 0x1000
  initial ramdisk...: /boot/initrd-2.6.7-1.451.2.3.img at 0x800000
Preparing boot device: dasda (0201).
Preparing boot menu
```

```
  Interactive prompt......: disabled
  Menu timeout............: disabled
  Default configuration...: 'linux'
Syncing disks...
Done.
```

22.6. Adding a Network Device

The process of adding a network device has changed with the migration of the 2.4 kernel to the 2.6 kernel:

- The **proc** file system is no longer used to control or obtain status on network devices.
- The new **sys** file system now provides facilities for controlling devices.
- **/sys/class/net/interface_name/device** now provides status on active devices.

  interface_name is a name such as **eth0** or **eth2** that is given to a network interface by the device driver when the device is configured.
- **/etc/chandev.conf** no longer exists.

  The **sys** file system now contains the information that was placed in **/etc/chandev.conf**.
- **/etc/modules.conf** no longer exists.

  Network interface alias specifications are now placed in **/etc/modprobe.conf**.

Section 22.6.1, “Adding a **qeth** Device” describes in detail how to add a qeth device to an existing instance of Red Hat Enterprise Linux. Section 22.6.2, “Quick Reference for Adding Network Devices” is a quick reference for installing other IBM System z network interfaces.

22.6.1. Adding a qeth Device

First, determine whether the **qeth** device driver modules are loaded.

```
# lsmod | grep qeth
qeth                  135240  0
qdio                   45360  2 qeth
ipv6                  303984  13 qeth
ccwgroup               15104  1 qeth
```

If the output of the **lsmod** command shows that the modules are not loaded, you must run the **modprobe** command to load them:

```
# modprobe qeth
```

Next, create a **qeth** group device.

```
# echo read_device_bus_id,write_device_bus_id,
data_device_bus_id > /sys/bus/ccwgroup/drivers/qeth/group
```

Due to the length of this command, it has been broken into two lines.

In the following example, *read_device_bus_id* is 0.0.0600, *write_device_bus_id* is 0.0.0601, and *data_device_bus_id* is 0.0.0602. The device is a z/VM virtual NIC and the IP address to be assigned to this interface is 192.168.70.69.

```
# echo 0.0.0600,0.0.0601,0.0.0602 > /sys/bus/ccwgroup/drivers/qeth/group
```

Next, verify that the **qeth** group device was created properly:

```
# ls /sys/bus/ccwgroup/drivers/qeth
0.0.0600  0.0.09a0  group  notifier_register
```

You may optionally add a portname. First, you must check to see if a portname is required:

```
# cat /sys/bus/ccwgroup/drivers/qeth/0.0.0600/portname
no portname required
```

The response indicates that you do not need to provide a portname.

To add a port name, check that the devices are offline, and then run the following command:

Note

The device(s) must be offline when you add a portname.

```
# echo portname > /sys/bus/ccwgroup/drivers/qeth/0.0.0600/portname
```

Next, bring the device back online:

```
# echo 1 > /sys/bus/ccwgroup/drivers/qeth/0.0.0600/online
```

Then verify the state of the device:

```
# cat /sys/bus/ccwgroup/drivers/qeth/0.0.0600/online
1
```

A return value of "1" indicates that the device is online, while a return value '0' indicates that the device is offline.

Check to see what interface name was assigned to the device:

```
# cat /sys/bus/ccwgroup/drivers/qeth/0.0.0600/if_name
eth1
```

You may optionally set additional parameters and features, depending on the way you are setting up your system and the features you require.

- ***add_hhlen***
- ***broadcast_mode***
- ***buffer_count***
- ***canonical_macaddr***
- ***checksumming***
- ***detach_state***
- ***fake_broadcast***

- ***fake_ll***
- ***ipa_takeover***
- ***portno***
- ***priority_queueing***
- ***recover***
- ***route4***
- ***rxip***
- ***ungroup***
- ***vipa***

For information on how these features work, refer to http://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.wss?CTY=US&FNC=SRX&PBL=SC33-8289-02 (*Device Drivers, Features, and Commands - SC33-8289-02*).

Now you need to create the configuration file for your new interface. The network interface configuration files are placed in **/etc/sysconfig/network-scripts/**.

The network configuration files use the naming convention ifcfg-*device*, where device is the value found in the **if_name** file in the qeth group device that was created earlier. In this example it is **eth1**.

If there is an existing configuration file for another device of the same type already defined, the simplest solution is to copy it to the new name.

```
# cd /etc/sysconfig/network-scripts
# cp ifcfg-eth0 ifcfg-eth1
```

If you do not have a similar device defined you must create one. Use this example of **ifcfg-eth0** as a template.

```
/etc/sysconfig/network-scripts/ifcfg-eth0
# IBM QETH
DEVICE=eth0
BOOTPROTO=static
HWADDR=00:06:29:FB:5F:F1
IPADDR=9.12.20.136
NETMASK=255.255.255.0
ONBOOT=yes
NETTYPE=qeth
SUBCHANNELS=0.0.09a0,0.0.09a1,0.0.09a2
TYPE=Ethernet
```

Edit the new **ifcfg-eth1** file.

Remove the HWADDR line for now.

Modify the DEVICE statement to reflect the contents of the **if_name** file from your ccwgroup.

Modify the IPADDR statement to reflect the IP address of your new interface.

Modify the NETMASK statement as needed.

If you want your new interface to be activated at boot time, then make sure ONBOOT is set to ***yes*** .

Make sure the SUBCHANNELS statement matches the hardware addresses for your **qeth** device.

```
/etc/sysconfig/network-scripts/ifcfg-eth1
# IBM QETH
DEVICE=eth1
BOOTPROTO=static
IPADDR=192.168.70.87
NETMASK=255.255.255.0
ONBOOT=yes
NETTYPE=qeth
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
TYPE=Ethernet
```

A **qeth** device requires an alias definition in **/etc/modprobe.conf**. Edit this file and add an alias for your interface.

```
/etc/modprobe.conf
alias eth0 qeth
alias eth1 qeth
options dasd_mod dasd=0.0.0100,0.0.4b19
```

Now you can start the new interface:

```
# ifup eth1
```

Check the status of the interface:

```
# ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 02:00:00:00:00:01
          inet addr:192.168.70.87  Bcast:192.168.70.255  Mask:255.255.255.0
          inet6 addr: fe80::ff:fe00:1/64 Scope:Link
          UP BROADCAST RUNNING NOARP MULTICAST  MTU:1492  Metric:1
          RX packets:23 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:644 (644.0 b)  TX bytes:264 (264.0 b)
```

Note that the **HWaddr** field in the first line of the **ifconfig** command output. The value after that must be added to the **ifcfg-eth1** file. Add a line like the following to that file:

```
HWADDR=02:00:00:00:00:01
```

Now **ifcfg-eth1** looks similar to the following:

```
# IBM QETH
DEVICE=eth1
HWADDR=02:00:00:00:00:01
BOOTPROTO=static
IPADDR=192.168.70.69
NETMASK=255.255.255.0
ONBOOT=yes
NETTYPE=qeth
SUBCHANNELS=0.0.0600,0.0.0601,0.0.0602
TYPE=Ethernet
```

Check the routing for the new interface:

```
# route
Kernel IP routing table
```

```
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.70.0    *               255.255.255.0   U     0      0        0 eth1
9.12.20.0       *               255.255.255.0   U     0      0        0 eth0
169.254.0.0     *               255.255.0.0     U     0      0        0 eth1
default         pdlrouter-if5.p 0.0.0.0         UG    0      0        0 eth0
```

Verify your changes by using the **ping** command to ping the gateway:

```
# ping -c 1 192.168.70.8
PING 192.168.70.8 (192.168.70.8) 56(84) bytes of data.
64 bytes from 192.168.70.8: icmp_seq=0 ttl=63 time=8.07 ms
```

If the default route information has changed, you must also update **/etc/sysconfig/network** accordingly.

22.6.2. Quick Reference for Adding Network Devices

There are several basic tasks for adding a network interface on IBM System z.

- Load the device driver.
- Create the group device.
- Configure the device.
- Set the device online.
- Define the alias (if needed).
- Create a configuration script.
- Activate the device.

The following sections provide basic information for each task of each IBM System z network device driver.

22.6.2.1. Working With the LCS Device Driver

The LAN channel station (LCS) device driver supports OSA-2 Ethernet/Token Ring, OSA-Express Fast Ethernet in non-QDIO mode, and OSA-Express High Speed Token Ring in non-QDIO mode. For z990, the LCS driver also supports Gigabit Ethernet in non-QDIO mode (including 1000Base-T).

Based on the type of interface being added, the LCS driver assigns one of two base interface names: eth*n* for OSA-Express Fast Ethernet and Gigabit Ethernet tr*n* for Token Ring, where *n* is an integer that uniquely identifies the device. *n* is 0 for the first device of that type, 1 for the second, and so on.

- Load the device driver:

  ```
  # modprobe lcs
  ```

- Create the group device:

  ```
  # echo read_device_bus_id,write_device_bus_id >
    /sys/bus/ccwgroup/drivers/lcs/group
  ```

  Due to the length of this command, it has been broken into two lines.

- Configure the device.

OSA cards can provide up to 16 ports for a single CHPID. By default, the LCS group device uses port 0. To use a different port, issue a command similar to the following:

```
# echo portno > /sys/bus/ccwgroup/drivers/lcs/device_bus_id/portno
```

For more information about configuration of the LCS driver, refer to the following:

http://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.wss?CTY=US&FNC=SRX&PBL=SC33-8289-02 (Linux for IBM System z and S/390 Device Drivers, Features, and Commands)

- Set the device online:

```
# echo 1 > /sys/bus/ccwgroup/drivers/lcs/read_device_bus_id/online
```

- Define the alias.

  Based on the type interface being added, add a line to **/etc/modprobe.conf** that is similar to one of the following:

```
ethn alias lcs
trn alias lcs
```

- Create a configuration script.

  Create a file in **/etc/sysconfig/network-scripts/** with a name like one of the following:

```
ifcfg-ethn
ifcfg-trn
```

The file should look similar to the following:

```
/etc/sysconfig/network-scripts/ifcfg-eth0
# IBM LCS
DEVICE=eth0
BOOTPROTO=static
HWADDR=00:06:29:FB:5F:F1
IPADDR=9.12.20.136
NETMASK=255.255.255.0
ONBOOT=yes
NETTYPE=lcs
SUBCHANNELS=0.0.09a0,0.0.09a1
PORTNAME=0
TYPE=Ethernet
```

Based on the type interface being added, the DEVICE parameter should be one of the following:

```
DEVICE=ethn
DEVICE=trn
```

- Activate the device.

  Based on the type interface being added, issue an **ifup** command:

```
# ifup ethn
# ifup trn
```

22.6.2.2. Working With the QETH Device Driver

The QETH network device driver supports IBM System z HiperSockets, OSA-Express Fast Ethernet, Gigabit Ethernet (including 1000Base-T), High Speed Token Ring, and ATM features (running Ethernet LAN emulation) in QDIO mode.

Based on the type of interface being added, the QETH driver assigns one of three base interface names:

- hsi*n* for HiperSocket devices
- eth*n* for OSA-Express Fast Ethernet and Gigabit Ethernet
- tr*n* for Token Ring

The value *n* is an integer that uniquely identifies the device. *n* is 0 for the first device of that type, 1 for the second, and so on.

- Load the device driver:

```
# modprobe qeth
```

- Create the group device:

```
# echo read_device_bus_id,write_device_bus_id,data_device_bus_id >
/sys/bus/ccwgroup/drivers/qeth/group
```

  Due to the length of this command, it has been broken into two lines.

- Configure the device.

  For more information about configuration of the QETH driver, refer to the following:

  http://oss.software.ibm.com/developerworks/opensource/linux390/docu/lx26apr04dd01.pdf (Linux for IBM System z and S/390 Device Drivers, Features, and Commands)

- Set the device online:

```
# echo 1 > /sys/bus/ccwgroup/drivers/qeth/read_device_bus_id/online
```

- Define the alias.

  Based on the type interface being added, add a line to **/etc/modprobe.conf** that is like one of the following:

```
hsin alias qeth
ethn alias qeth
trn alias qeth
```

- Create a configuration script.

  Create a file in **/etc/sysconfig/network-scripts/** with a name like one of the following:

```
ifcfg-hsin
ifcfg-ethn
ifcfg-trn
```

  The file should look like this:

```
/etc/sysconfig/network-scripts/ifcfg-eth0
# IBM QETH
```

```
DEVICE=eth0
BOOTPROTO=static
HWADDR=00:06:29:FB:5F:F1
IPADDR=9.12.20.136
NETMASK=255.255.255.0
ONBOOT=yes
NETTYPE=qeth
SUBCHANNELS=0.0.09a0,0.0.09a1,0.0.09a2
TYPE=Ethernet
```

Based on the type interface being added, the DEVICE parameter should be like one of the following:

```
DEVICE=hsin
DEVICE=ethn
DEVICE=trn
```

- Activate the device.

  Based on the type interface being added, issue an **ifup** command:

```
# ifup hsin
# ifup ethn
# ifup trn
```

22.7. Kernel-Related Information

Red Hat Enterprise Linux includes a modification to the way the Linux kernel timer interrupt is handled. Normally, a hardware timer is set to generate periodic interrupts at a fixed rate (100 times a second for most architectures). These periodic timer interrupts are used by the kernel to schedule various internal housekeeping tasks, such as process scheduling, accounting, and maintaining system uptime.

While a timer-based approach works well for a system environment where only one copy of the kernel is running, it can cause additional overhead when many copies of the kernel are running on a single system (for example, as z/VM(R) guests). In these cases, having thousands of copies of the kernel each generating interrupts many times a second can result in excessive system overhead.

Therefore, Red Hat Enterprise Linux now includes the ability to turn off periodic timer interrupts. This is done through the **/proc/** file system. To disable periodic timer interrupts, issue the following command:

```
echo 0 > /proc/sys/kernel/hz_timer
```

To enable periodic timer interrupts, issue the following command:

```
echo 1 > /proc/sys/kernel/hz_timer
```

By default, periodic timer interrupts are disabled.

Periodic timer interrupt states can also be set at boot-time; to do so, add the following line to **/etc/sysctl.conf** to disable periodic timer interrupts:

```
kernel.hz_timer = 0
```

Note

Disabling periodic timer interrupts can violate basic assumptions in system accounting tools. If you notice a malfunction related to system accounting, verify that the malfunction disappears if periodic timer interrupts are enabled, then submit a bug at http://bugzilla.redhat.com/bugzilla/ (for malfunctioning bundled tools), or inform the tool vendor (for malfunctioning third-party tools).

Part IV. Common Tasks

Information that is common to all architectures related to registering your system with Red Hat Network, choosing whether to install or upgrade, and information on disk partitioning is contained in this section.

Chapter 23. Updating Your System

23.1. Driver update rpm packages

Occasionally, when a new piece of hardware is not yet supported in the kernel that you have installed, Red Hat or a hardware vendor might make a driver update available. Although you can install driver updates during the installation process (refer to *Chapter 7, Updating drivers during installation on Intel and AMD systems* for Intel and AMD systems and *Chapter 13, Updating drivers during installation on IBM POWER systems* for IBM POWER systems) we recommend that you do this only for devices that are essential to carry out the installation. In all other cases, complete the installation first, and then add support for the device with a driver update rpm package as described in this section.

Do not install a driver update rpm unless you are certain that your system requires it. Installing a driver update on a system for which it was not intended can cause system difficulties.

To see a list of driver updates already installed on your system, click **Applications > Add/Remove Software** on your desktop, and enter the root password if prompted for it. Click the **Search** tab, enter the word **kmod-** (notice the final **-**) and click **Search**.

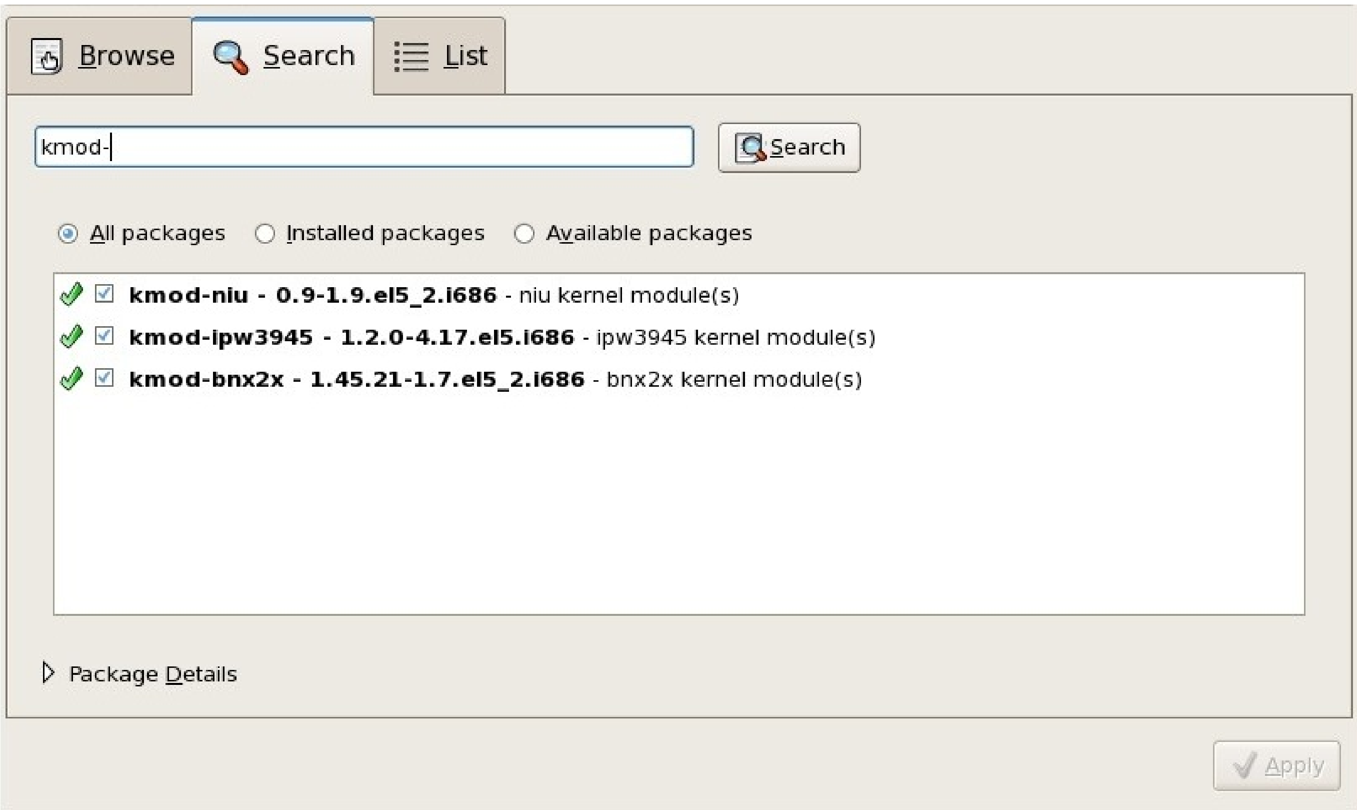

Figure 23.1. Listing installed Driver Update RPM packages

Alternatively, you can use the command line, as follows:

```
$ rpm -qa | egrep ^kmod-
```

Note the **-** on the end of **kmod**. This will list all installed packages that begin with **kmod-**, which should include all driver updates that are currently installed on your system. Additional drivers provided by third-party update software are not listed in this output. Contact the third-party vendor for details.

To install a new driver update rpm package:

1. Download the driver update rpm package from the location specified by Red Hat or your hardware vendor. The package file name will begin with **kmod** (short for *kernel module*) and have a form similar to this example:

   kmod-ipw3945-1.2.04.17.el5.i686.rpm

   In the example, the driver update rpm package supplies an Intel IPW3945 WiFi driver update with version number 1.2.0-4.17 for Red Hat Enterprise Linux 5, on i686 systems. A version of this driver package for systems running a Xen kernel will be similar, but include **xen** in the package name:

   kmod-ipw3945-xen-1.2.04.17.el5.i686.rpm

   Driver update rpm packages are signed packages, and like all other software packages, they are automatically validated at install time. To perform this step manually, type the following at a command line:

   ```
   $ rpm --checksig -v filename.rpm
   ```

   where *filename.rpm* is the driver update rpm package file name. This verifies the package against using the standard Red Hat GPG package signing key that is already installed on any Red Hat Enterprise Linux 5.11 system. If you need this key for verification purposes on another system, you can can obtain it from: https://www.redhat.com/security/team/key/

2. Locate and double-click the file that you downloaded. The system might prompt you for the root password, after which it will present the following **Installing Packages** box:

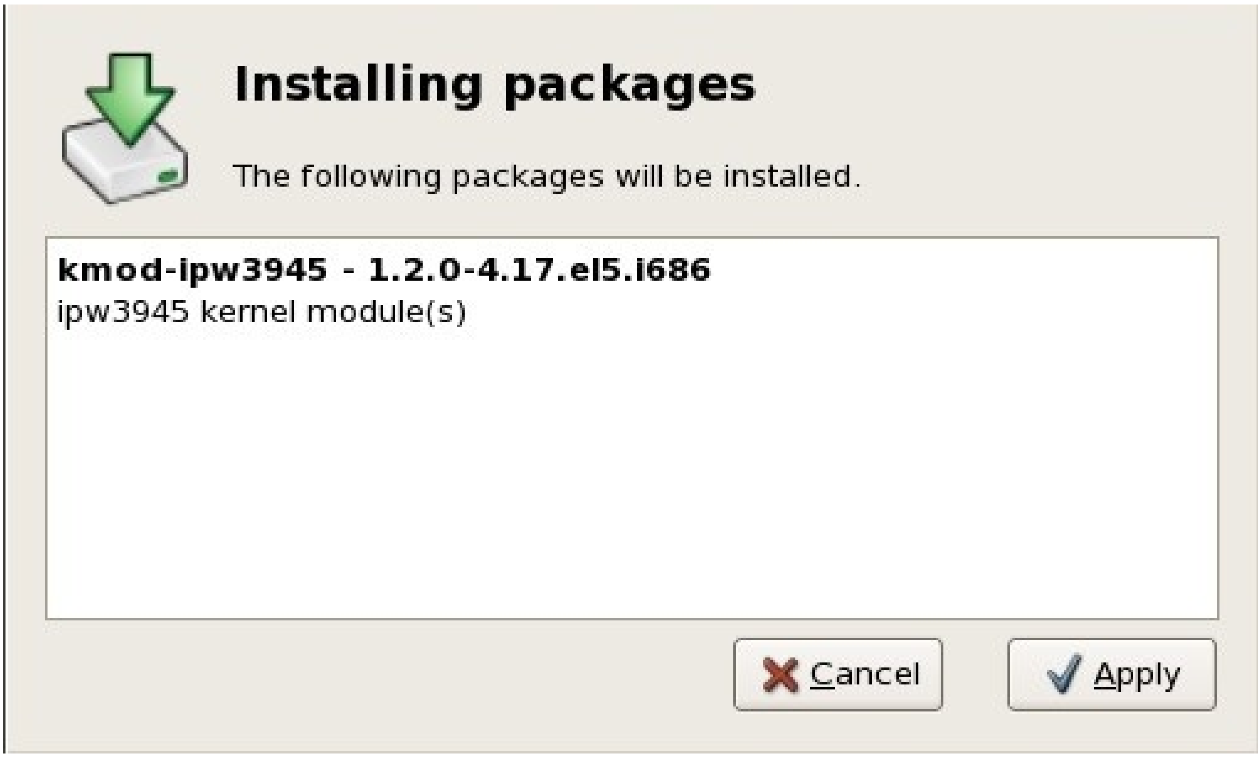

Figure 23.2. The installing packages box

   Click **Apply** to complete the package installation.

   Alternatively, you can install a driver update manually on the command line:

```
$ rpm -ivh kmod-ipw3945-1.2.04.17.el5.i686
```

3. Whether you used a graphical install, or a command line install, reboot your system to ensure your system is using the new driver.

If Red Hat ships a kernel errata update before the next release of Red Hat Enterprise Linux, your system will continue to use the driver updates that you have installed. There is no need to re-install driver updates following an errata update. Generally, when Red Hat releases a new version of Red Hat Enterprise Linux, all driver updates for the previous version are incorporated in the new version. However, if it was not possible to include a particular driver, you will need to perform another driver update when you install the new version of Red Hat Enterprise Linux. In this case, Red Hat or your hardware party vendor will inform you of the location of the update.

Chapter 24. Upgrading Your Current System

This chapter explains the various methods available for upgrading your Red Hat Enterprise Linux system.

24.1. Determining Whether to Upgrade or Re-Install

Red Hat does not support upgrades from earlier major versions

Although **anaconda** provides an option for upgrading from earlier major versions of Red Hat Enterprise Linux to Red Hat Enterprise Linux 5.11, Red Hat does not currently support this. More generally, Red Hat does not support in-place upgrades between any major versions of Red Hat Enterprise Linux. (A major version is denoted by a whole number version change. For example, Red Hat Enterprise Linux 4 and Red Hat Enterprise Linux 5 are both major versions of Red Hat Enterprise Linux.)

In-place upgrades across major releases do not preserve all system settings, services or custom configurations. Consequently, Red Hat strongly recommends fresh installations when upgrading from one major version to another.

While upgrading from Red Hat Enterprise Linux version 4 Update 4 is technically possible, you are more likely to have a consistent experience by backing up your data and then installing this release of Red Hat Enterprise Linux 5.11 over your previous Red Hat Enterprise Linux installation.

To upgrade from Red Hat Enterprise Linux 4 you should bring your system up to date using RHN before performing the upgrade.

This recommended reinstallation method helps to ensure the best system stability possible.

For more information about re-installing your Red Hat Enterprise Linux system, refer to the Whitepapers available online at http://www.redhat.com/rhel/resource_center/.

If you currently use Red Hat Enterprise Linux 4 Update 4, you can perform a traditional, installation program-based upgrade.

However, before you choose to upgrade your system, there are a few things you should keep in mind:

- Individual package configuration files may or may not work after performing an upgrade due to changes in various configuration file formats or layouts.
- Third party or ISV applications may not work correctly following the upgrade.
- If you have one of Red Hat's layered products (such as the Cluster Suite) installed, it may need to be manually upgraded after the Red Hat Enterprise Linux upgrade has been completed.
- If you have package repositories from providers other than Red Hat enabled, note that software installed from those repositories may not function properly after a system upgrade. Red Hat cannot guarantee that such repositories are up-to-date.

Upgrading your system installs updated versions of the packages which are currently installed on your system.

The upgrade process preserves existing configuration files by renaming them with an **.rpmsave** extension (for example, **sendmail.cf.rpmsave**). The upgrade process also creates a log of its actions in **/root/upgrade.log**.

Warning

As software evolves, configuration file formats can change. It is very important to carefully compare your original configuration files to the new files before integrating your changes.

Note

It is always a good idea to back up any data that you have on your systems. For example, if you are upgrading or creating a dual-boot system, you should back up any data you wish to keep on your hard drive(s). Mistakes do happen and can result in the loss of all of your data.

Some upgraded packages may require the installation of other packages for proper operation. If you choose to customize your packages to upgrade, you may be required to resolve dependency problems. Otherwise, the upgrade procedure takes care of these dependencies, but it may need to install additional packages which are not on your system.

Depending on how you have partitioned your system, the upgrade program may prompt you to add an additional swap file. If the upgrade program does not detect a swap file that equals twice your RAM, it asks you if you would like to add a new swap file. If your system does not have a lot of RAM (less than 256 MB), it is recommended that you add this swap file.

24.2. Upgrading Your System

The **Upgrade Examine** screen appears if you have instructed the installation program to perform an upgrade.

Note

If the contents of your **/etc/redhat-release** file have been changed from the default, your Red Hat Enterprise Linux installation may not be found when attempting an upgrade to Red Hat Enterprise Linux 5.11.

You can relax some of the checks against this file by booting with the following boot command:

```
linux upgradeany
```

Use the **linux upgradeany** command if your Red Hat Enterprise Linux installation was not given as an option to upgrade.

To perform an upgrade, select **Perform an upgrade of an existing installation**. Click **Next** when you are ready to begin your upgrade.

To re-install your system, select **Perform a new Red Hat Enterprise Linux installation** and refer to http://www.redhat.com/docs/wp/ as well as Chapter 4, *Installing on Intel® and AMD Systems*, Chapter 12, *Installing on IBM System i and IBM System p systems*, or Chapter 17, *Installing on IBM System z Systems* for further instructions.

To perform a new installation of Red Hat Enterprise Linux on your system, select **Perform a new Red Hat Enterprise Linux installation** and refer to Chapter 4, *Installing on Intel® and AMD Systems*, Chapter 12, *Installing on IBM System i and IBM System p systems*, or Chapter 17, *Installing on IBM System z Systems* for further instructions.

Chapter 25. Register Your System and Apply Subscriptions

25.1. Registering the System

Before you can access service and software maintenance information and enhanced support included as part of your subscription, you must register your new system with Red Hat and associate or *attach* the appropriate subscriptions with your system.

A Red Hat subscription supplied your system access to:

- Software updates, errata, and maintenance
- Red Hat technical support resources and Knowledgebase articles

The Red Hat Subscription Management Subscription Concepts and Workflows has information on different environments and workflows. The basic registration process (which is described in this section) registers a system with Customer Portal Subscription Management (Red Hat's hosted services) and automatically subscribes the system to the best-matched subscriptions. It is also possible to register a system with a Subscription Asset Manager subscription service, CloudForms System Engine, or to register a disconnected system through the Customer Portal.

25.1.1. Registering at Firstboot

The first time you boot a new Red Hat Enterprise Linux system, the **firstboot** system prompts you to register the system and auto-attach the matching subscriptions.

Red Hat uses *X.509 certificates* to identify installed products on a system, the subscriptions attached to a system, and the system itself within the subscription service inventory. There are several different subscription services which use and recognize certificate-based subscriptions, and a system can be registered with any of them in firstboot:

- Customer Portal Subscription Management, hosted services from Red Hat (the default)
- Subscription Asset Manager, an on-premise subscription server which proxies content delivery back to the Customer Portal's services
- CloudForms System Engine, an on-premise service which handles both subscription services and content delivery

The specific type of subscription/content service does not need to be selected; all three server types (Customer Portal Subscription Management, Subscription Asset Manager, and CloudForms System Engine) are within Red Hat Subscription Management and use the same types of service APIs. The only thing that needs to be identified is the hostname of the service to connect to and then the appropriate user credentials for that service.

1. To identify which subscription server to use for registration, enter the hostname of the service. The default service is Customer Portal Subscription Management, with the hostname **`subscription.rhn.redhat.com`**. To use a different subscription service, such as Subscription Asset Manager, enter the hostname of the local server.
2. Click **`Forward`**.
3. Enter the user credentials **for the given subscription service** to log in.

> **Important**
>
> The user credentials to use depend on the subscription service. When registering with the Customer Portal, use the Red Hat Network credentials for the administrator or company account.
>
> However, for Subscription Asset Manager or CloudForms System engine, the user account to use is created within the on-premise service and probably is not the same as the Customer Portal user account.

If you have lost your login or password for the Customer Portal, recover them from https://www.redhat.com/wapps/sso/rhn/lostPassword.html. For lost login or password information for Subscription Asset Manager or CloudForms System Engine, contact your local administrator.

4. Set the system name for the host. This is anything which uniquely and clearly identifies the system within the subscription service inventory. This is usually the hostname or fully-qualified domain name of the machine, but it can be any string.

5. *Optional.* Set whether subscriptions should be set manually after registration. By default, this checkbox is unchecked so that the best-matched subscriptions are automatically applied to the system. Selecting this checkbox means that subscriptions must be added to the system manually after firstboot registration is complete. (Even if subscriptions are auto-attached, additional subscriptions can be added to the system later using the local Subscription Manager tools.)

6. When registration begins, firstboot scans for organizations and environments (sub-domains within the organization) to which to register the system.

   IT environments that use Customer Portal Subscription Management have only a single organization, so no further configuration is necessary. IT infrastructures that use a local subscription service like Subscription Asset Manager might have multiple organizations configured, and those organizations may have multiple environments configured within them.

   If multiple organizations are detected, Subscription Manager prompts to select the one to join.

7. If you decided to let Subscription Manager automatically attach subscriptions to the system (the default), then the system scans for the subscriptions to attach as part of the registration process.

   When registration is complete, the Subscription Manager reports the applied service level for the system based on the information in the selected subscription and the specific subscription that has been attached to the new system. This subscription selection must be confirmed to complete the registration process.

   If you selected to apply subscriptions later, then that part of the registration process is skipped, and the Subscription Manager screen in firstboot simply instructs you to attach subscriptions later.

8. Click **Forward** to move to the next configuration area for firstboot, user setup.

25.1.2. Registering After Firstboot

The system can be registered using the local Red Hat Subscription Manager tools.

Note

Red Hat Subscription Manager (GUI and CLI) must be run as root.

To register from the command line, use the **register** command with the **--autosubscribe** option so that the best-matched subscriptions are automatically attached. For example:

```
[root@server ~]# subscription-manager register --autosubscribe
Username: admin@example.com
Password:

The system has been registered with id: 30a3dc1b-db07-4ee7-bfb0-e09504b4033c
Installed Product Current Status:
Product Name:           Red Hat Enterprise Linux Server
Status:                 Subscribed
```

To register using the Red Hat Subscription Manager GUI:

1. Launch Subscription Manager. For example:

```
[root@server ~]# subscription-manager-gui
```

2. In the **System** menu of the Subscription Manager window, select the **Register** item.

3. Enter the hostname of the subscription server to which to register the system. This server, by default, is Customer Portal Subscription Management (the hosted service), at **subscription.rhn.redhat.com**. To register with a Subscription Asset Manager server or a CloudForms System Engine server, enter the appropriate hostname.

4. Enter the username and password of the user account on the subscription service.

Important

The user credentials to use depend on the subscription service. When registering with the Customer Portal, use the Red Hat Network credentials for the administrator or company account.

However, for Subscription Asset Manager or CloudForms System engine, the user account to use is created within the on-premise service and probably is not the same as the Customer Portal user account.

5. Optionally, select the **Skip automatic subscription selection...** checkbox.

   By default, the registration process automatically subscribes the system to the best matched subscription. This can be turned off so that the subscriptions can be selected manually.

25.1.3. Unregistering the System

The system is unregistered from a Red Hat Subscription Management service — Customer Portal Subscription Management, Subscription Asset Manager, or CloudForms System Engine using the Red Hat Subscription Manager tools.

For example, from the command line, use the **unregister** command:

```
[root@server1 ~]# subscription-manager unregister
```

From the Subscription Manager UI:

1. Launch Subscription Manager. For example:

   ```
   [root@server ~]# subscription-manager-gui
   ```

2. In the **System** menu of the Subscription Manager window, select the **Unregister** item.

Chapter 26. An Introduction to Disk Partitions

Note

This appendix is not necessarily applicable to non-x86-based architectures. However, the general concepts mentioned here may apply.

This appendix is not necessarily applicable to non-x86-based architectures. However, the general concepts mentioned here may apply.

If you are reasonably comfortable with disk partitions, you could skip ahead to Section 26.1.4, "Making Room For Red Hat Enterprise Linux", for more information on the process of freeing up disk space to prepare for a Red Hat Enterprise Linux installation. This section also discusses the partition naming scheme used by Linux systems, sharing disk space with other operating systems, and related topics.

26.1. Hard Disk Basic Concepts

Hard disks perform a very simple function — they store data and reliably retrieve it on command.

When discussing issues such as disk partitioning, it is important to know a bit about the underlying hardware. Unfortunately, it is easy to become bogged down in details. Therefore, this appendix uses a simplified diagram of a disk drive to help explain what is really happening when a disk drive is partitioned. Figure 26.1, "An Unused Disk Drive", shows a brand-new, unused disk drive.

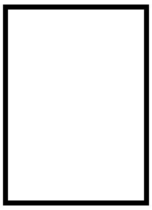

Figure 26.1. An Unused Disk Drive

Not much to look at, is it? But if we are talking about disk drives on a basic level, it is adequate. Say that we would like to store some data on this drive. As things stand now, it will not work. There is something we need to do first.

26.1.1. It is Not What You Write, it is How You Write It

Experienced computer users probably got this one on the first try. We need to *format* the drive. Formatting (usually known as "making a *file system*") writes information to the drive, creating order out of the empty space in an unformatted drive.

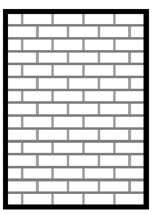

Figure 26.2. Disk Drive with a File System

As Figure 26.2, “Disk Drive with a File System”, implies, the order imposed by a file system involves some trade-offs:

- A small percentage of the drive's available space is used to store file system-related data and can be considered as overhead.
- A file system splits the remaining space into small, consistently-sized segments. For Linux, these segments are known as *blocks*. [11]

Given that file systems make things like directories and files possible, these trade-offs are usually seen as a small price to pay.

It is also worth noting that there is no single, universal file system. As Figure 26.3, “Disk Drive with a Different File System”, shows, a disk drive may have one of many different file systems written on it. As you might guess, different file systems tend to be incompatible; that is, an operating system that supports one file system (or a handful of related file system types) may not support another. This last statement is not a hard-and-fast rule, however. For example, Red Hat Enterprise Linux supports a wide variety of file systems (including many commonly used by other operating systems), making data interchange between different file systems easy.

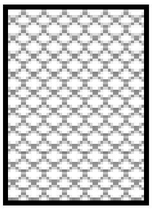

Figure 26.3. Disk Drive with a Different File System

Of course, writing a file system to disk is only the beginning. The goal of this process is to actually *store* and *retrieve* data. Let us take a look at our drive after some files have been written to it.

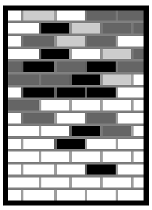

Figure 26.4. Disk Drive with Data Written to It

As Figure 26.4, “Disk Drive with Data Written to It”, shows, some of the previously-empty blocks are now holding data. However, by just looking at this picture, we cannot determine exactly how many files reside on this drive. There may only be one file or many, as all files use at least one block and some files use multiple blocks. Another important point to note is that the used blocks do not have to form a contiguous region; used and unused blocks may be interspersed. This is known as *fragmentation*. Fragmentation can play a part when attempting to resize an existing partition.

As with most computer-related technologies, disk drives changed over time after their introduction. In particular, they got bigger. Not larger in physical size, but bigger in their capacity to store information. And, this additional capacity drove a fundamental change in the way disk drives were used.

26.1.2. Partitions: Turning One Drive Into Many

As disk drive capacities soared, some people began to wonder if having all of that formatted space in one big chunk was such a great idea. This line of thinking was driven by several issues, some philosophical, some technical. On the philosophical side, above a certain size, it seemed that the additional space provided by a larger drive created more clutter. On the technical side, some file systems were never designed to support anything above a certain capacity. Or the file systems *could* support larger drives with a greater capacity, but the overhead imposed by the file system to track files became excessive.

The solution to this problem was to divide disks into *partitions*. Each partition can be accessed as if it was a separate disk. This is done through the addition of a *partition table*.

> **Note**
>
> While the diagrams in this chapter show the partition table as being separate from the actual disk drive, this is not entirely accurate. In reality, the partition table is stored at the very start of the disk, before any file system or user data. But for clarity, they are separate in our diagrams.

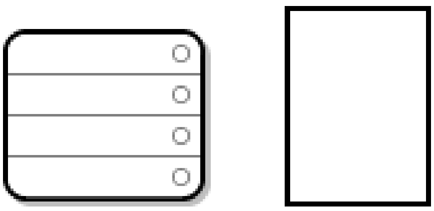

Figure 26.5. Disk Drive with Partition Table

As Figure 26.5, "Disk Drive with Partition Table" shows, the partition table is divided into four sections or four *primary* partitions. A primary partition is a partition on a hard drive that can contain only one logical drive (or section). Each section can hold the information necessary to define a single partition, meaning that the partition table can define no more than four partitions.

Each partition table entry contains several important characteristics of the partition:

- The points on the disk where the partition starts and ends
- Whether the partition is "active"
- The partition's type

Let us take a closer look at each of these characteristics. The starting and ending points actually define the partition's size and location on the disk. The "active" flag is used by some operating systems' boot loaders. In other words, the operating system in the partition that is marked "active" is booted.

The partition's type can be a bit confusing. The type is a number that identifies the partition's anticipated usage. If that statement sounds a bit vague, that is because the meaning of the partition type is a bit vague. Some operating systems use the partition type to denote a specific file system

type, to flag the partition as being associated with a particular operating system, to indicate that the partition contains a bootable operating system, or some combination of the three.

By this point, you might be wondering how all this additional complexity is normally used. Refer to Figure 26.6, “Disk Drive With Single Partition”, for an example.

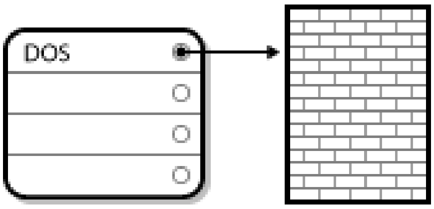

Figure 26.6. Disk Drive With Single Partition

In many cases, there is only a single partition spanning the entire disk, essentially duplicating the method used before partitions. The partition table has only one entry used, and it points to the start of the partition.

We have labeled this partition as being of the "DOS" type. Although it is only one of several possible partition types listed in Table 26.1, “Partition Types”, it is adequate for the purposes of this discussion.

Table 26.1, “Partition Types”, contains a listing of some popular (and obscure) partition types, along with their hexadecimal numeric values.

Table 26.1. Partition Types

Partition Type	Value	Partition Type	Value
Empty	00	Novell Netware 386	65
DOS 12-bit FAT	01	PIC/IX	75
XENIX root	02	Old MINIX	80
XENIX usr	03	Linux/MINUX	81
DOS 16-bit <=32M	04	Linux swap	82
Extended	05	Linux native	83
DOS 16-bit >=32	06	Linux extended	85
OS/2 HPFS	07	Amoeba	93
AIX	08	Amoeba BBT	94
AIX bootable	09	BSD/386	a5
OS/2 Boot Manager	0a	OpenBSD	a6
Win95 FAT32	0b	NEXTSTEP	a7
Win95 FAT32 (LBA)	0c	BSDI fs	b7
Win95 FAT16 (LBA)	0e	BSDI swap	b8
Win95 Extended (LBA)	0f	Syrinx	c7
Venix 80286	40	CP/M	db
Novell	51	DOS access	e1
PPC PReP Boot	41	DOS R/O	e3
GNU HURD	63	DOS secondary	f2
Novell Netware 286	64	BBT	ff

26.1.3. Partitions within Partitions — An Overview of Extended Partitions

Of course, over time it became obvious that four partitions would not be enough. As disk drives continued to grow, it became more and more likely that a person could configure four reasonably-sized partitions and still have disk space left over. There needed to be some way of creating more partitions.

Enter the extended partition. As you may have noticed in Table 26.1, “Partition Types”, there is an "Extended" partition type. It is this partition type that is at the heart of extended partitions.

When a partition is created and its type is set to "Extended," an extended partition table is created. In essence, the extended partition is like a disk drive in its own right — it has a partition table that points to one or more partitions (now called *logical partitions*, as opposed to the four *primary partitions*) contained entirely within the extended partition itself. Figure 26.7, “Disk Drive With Extended Partition”, shows a disk drive with one primary partition and one extended partition containing two logical partitions (along with some unpartitioned free space).

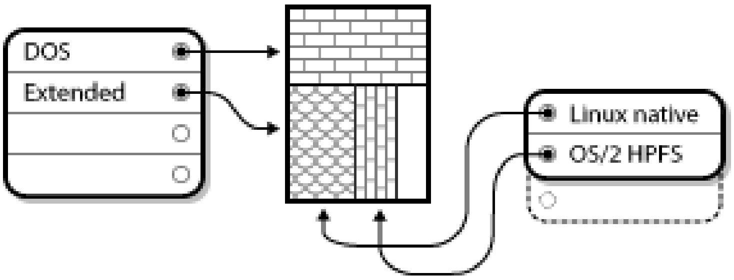

Figure 26.7. Disk Drive With Extended Partition

As this figure implies, there is a difference between primary and logical partitions — there can only be four primary partitions, but there is no fixed limit to the number of logical partitions that can exist. However, due to the way in which partitions are accessed in Linux, you should avoid defining more than 12 logical partitions on a single disk drive.

Now that we have discussed partitions in general, let us review how to use this knowledge to install Red Hat Enterprise Linux.

26.1.4. Making Room For Red Hat Enterprise Linux

The following list presents some possible scenarios you may face when attempting to repartition your hard disk:

- Unpartitioned free space is available
- An unused partition is available
- Free space in an actively used partition is available

Let us look at each scenario in order.

Note

Keep in mind that the following illustrations are simplified in the interest of clarity and do not reflect the exact partition layout that you encounter when actually installing Red Hat Enterprise Linux.

26.1.4.1. Using Unpartitioned Free Space

In this situation, the partitions already defined do not span the entire hard disk, leaving unallocated space that is not part of any defined partition. Figure 26.8, "Disk Drive with Unpartitioned Free Space", shows what this might look like.

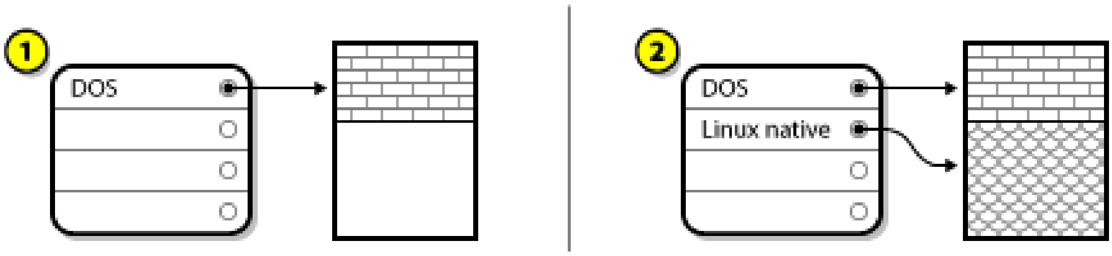

Figure 26.8. Disk Drive with Unpartitioned Free Space

In Figure 26.8, "Disk Drive with Unpartitioned Free Space", *1* represents an undefined partition with unallocated space and *2* represents a defined partition with allocated space.

If you think about it, an unused hard disk also falls into this category. The only difference is that *all* the space is not part of any defined partition.

In any case, you can create the necessary partitions from the unused space. Unfortunately, this scenario, although very simple, is not very likely (unless you have just purchased a new disk just for Red Hat Enterprise Linux). Most pre-installed operating systems are configured to take up all available space on a disk drive (refer to Section 26.1.4.3, "Using Free Space from an Active Partition").

Next, we will discuss a slightly more common situation.

26.1.4.2. Using Space from an Unused Partition

In this case, maybe you have one or more partitions that you do not use any longer. Perhaps you have dabbled with another operating system in the past, and the partition(s) you dedicated to it never seem to be used anymore. Figure 26.9, "Disk Drive With an Unused Partition", illustrates such a situation.

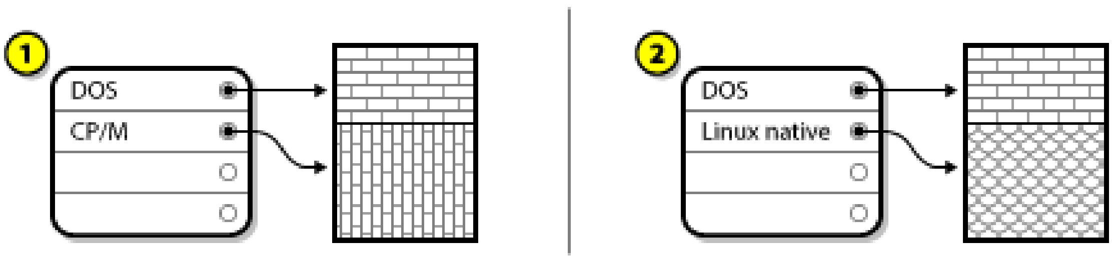

Figure 26.9. Disk Drive With an Unused Partition

In Figure 26.9, "Disk Drive With an Unused Partition", *1* represents an unused partition and *2* represents reallocating an unused partition for Linux.

If you find yourself in this situation, you can use the space allocated to the unused partition. You first must delete the partition and then create the appropriate Linux partition(s) in its place. You can delete the unused partition and manually create new partitions during the installation process.

26.1.4.3. Using Free Space from an Active Partition

This is the most common situation. It is also, unfortunately, the hardest to handle. The main problem is that, even if you have enough free space, it is presently allocated to a partition that is already in use. If you purchased a computer with pre-installed software, the hard disk most likely has one massive partition holding the operating system and data.

Aside from adding a new hard drive to your system, you have two choices:

Destructive Repartitioning

Basically, you delete the single large partition and create several smaller ones. As you might imagine, any data you had in the original partition is destroyed. This means that making a complete backup is necessary. For your own sake, make two backups, use verification (if available in your backup software), and try to read data from your backup *before* you delete the partition.

Warning

If there was an operating system of some type installed on that partition, it needs to be reinstalled as well. Be aware that some computers sold with pre-installed operating systems may not include the CD-ROM media to reinstall the original operating system. The best time to notice if this applies to your system is *before* you destroy your original partition and its operating system installation.

After creating a smaller partition for your existing operating system, you can reinstall any software, restore your data, and start your Red Hat Enterprise Linux installation. Figure 26.10, "Disk Drive Being Destructively Repartitioned" shows this being done.

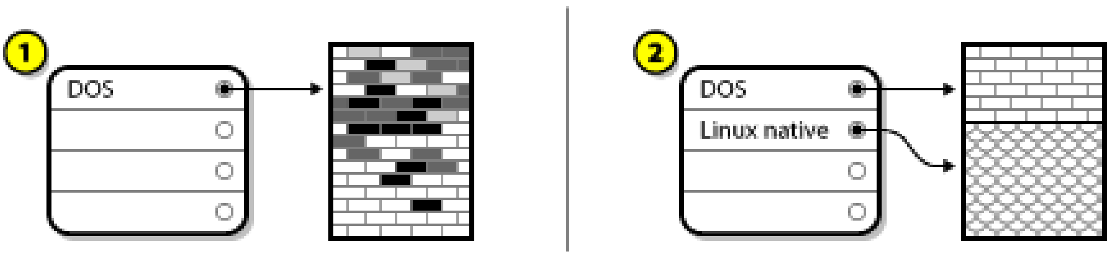

Figure 26.10. Disk Drive Being Destructively Repartitioned

In Figure 26.10, "Disk Drive Being Destructively Repartitioned", *1* represents before and *2* represents after.

Warning

As Figure 26.10, "Disk Drive Being Destructively Repartitioned", shows, any data present in the original partition is lost without proper backup!

Non-Destructive Repartitioning

Here, you run a program that does the seemingly impossible: it makes a big partition smaller without losing any of the files stored in that partition. Many people have found this method to be reliable and trouble-free. What software should you use to perform this feat?

There are several disk management software products on the market. Do some research to find the one that is best for your situation.

While the process of non-destructive repartitioning is rather straightforward, there are a number of steps involved:

- Compress and backup existing data
- Resize the existing partition
- Create new partition(s)

Next we will look at each step in a bit more detail.

26.1.4.3.1. Compress existing data

As Figure 26.11, “Disk Drive Being Compressed”, shows, the first step is to compress the data in your existing partition. The reason for doing this is to rearrange the data such that it maximizes the available free space at the "end" of the partition.

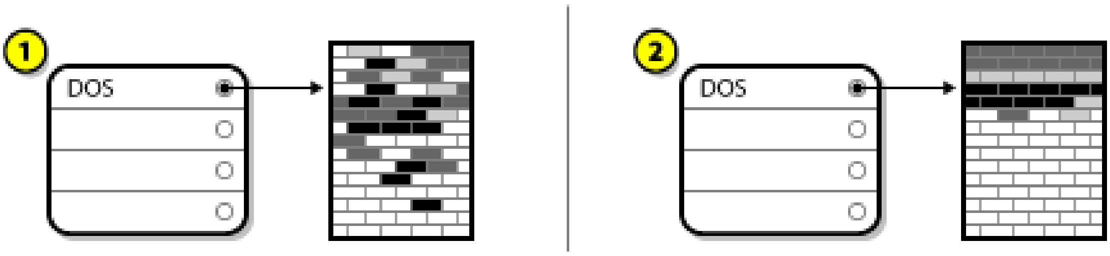

Figure 26.11. Disk Drive Being Compressed

In Figure 26.11, “Disk Drive Being Compressed”, *1* represents before and *2* represents after.

This step is crucial. Without it, the location of your data could prevent the partition from being resized to the extent desired. Note also that, for one reason or another, some data cannot be moved. If this is the case (and it severely restricts the size of your new partition(s)), you may be forced to destructively repartition your disk.

26.1.4.3.2. Resize the existing partition

Figure 26.12, “Disk Drive with Partition Resized”, shows the actual resizing process. While the actual result of the resizing operation varies depending on the software used, in most cases the newly freed space is used to create an unformatted partition of the same type as the original partition.

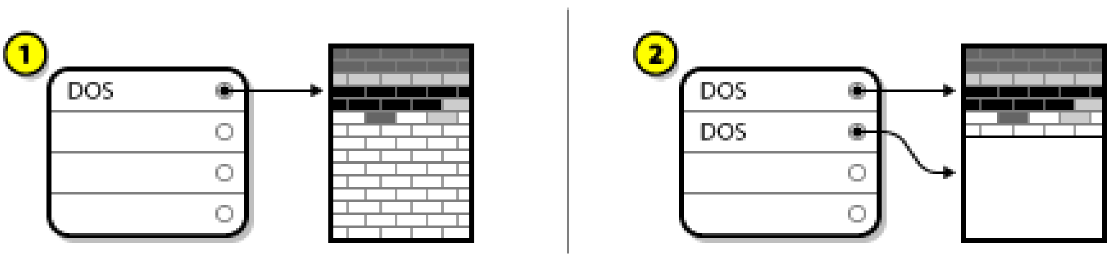

Figure 26.12. Disk Drive with Partition Resized

In Figure 26.12, “Disk Drive with Partition Resized”, *1* represents before and *2* represents after.

It is important to understand what the resizing software you use does with the newly freed space, so that you can take the appropriate steps. In the case we have illustrated, it would be best to delete the new DOS partition and create the appropriate Linux partition(s).

26.1.4.3.3. Create new partition(s)

As the previous step implied, it may or may not be necessary to create new partitions. However, unless your resizing software is Linux-aware, it is likely that you must delete the partition that was created during the resizing process. Figure 26.13, “Disk Drive with Final Partition Configuration”, shows this being done.

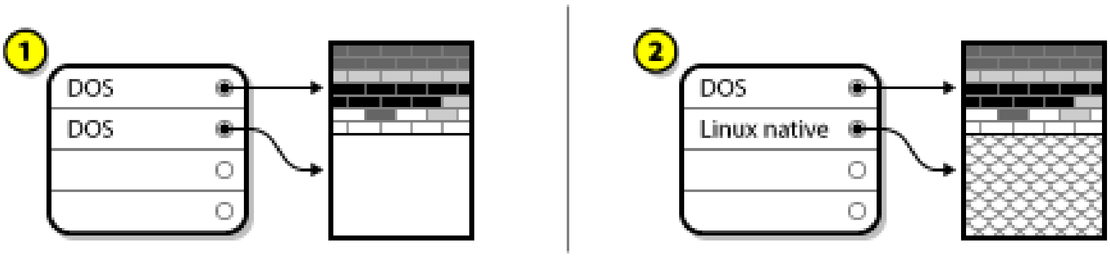

Figure 26.13. Disk Drive with Final Partition Configuration

In Figure 26.13, “Disk Drive with Final Partition Configuration”, *1* represents before and *2* represents after.

> **Note**
>
> The following information is specific to x86-based computers only.

As a convenience to our customers, we provide the **parted** utility. This is a freely available program that can resize partitions.

If you decide to repartition your hard drive with **parted**, it is important that you be familiar with disk storage and that you perform a backup of your computer data. You should make two copies of all the important data on your computer. These copies should be to removable media (such as tape, CD-ROM, or diskettes), and you should make sure they are readable before proceeding.

Should you decide to use **parted**, be aware that after **parted** runs you are left with *two* partitions: the one you resized, and the one **parted** created out of the newly freed space. If your goal is to use that space to install Red Hat Enterprise Linux, you should delete the newly created partition, either by using the partitioning utility under your current operating system or while setting up partitions during installation.

26.1.5. Partition Naming Scheme

Linux refers to disk partitions using a combination of letters and numbers which may be confusing, particularly if you are used to the "C drive" way of referring to hard disks and their partitions. In the DOS/Windows world, partitions are named using the following method:

- Each partition's type is checked to determine if it can be read by DOS/Windows.
- If the partition's type is compatible, it is assigned a "drive letter." The drive letters start with a "C" and move on to the following letters, depending on the number of partitions to be labeled.

- The drive letter can then be used to refer to that partition as well as the file system contained on that partition.

Red Hat Enterprise Linux uses a naming scheme that is more flexible and conveys more information than the approach used by other operating systems. The naming scheme is file-based, with file names in the form of **/dev/*xxyN***.

Here is how to decipher the partition naming scheme:

/dev/

This is the name of the directory in which all device files reside. Since partitions reside on hard disks, and hard disks are devices, the files representing all possible partitions reside in **/dev/**.

xx

The first two letters of the partition name indicate the type of device on which the partition resides, usually either **hd** (for IDE disks) or **sd** (for SCSI disks).

y

This letter indicates which device the partition is on. For example, **/dev/hda** (the first IDE hard disk) or **/dev/sdb** (the second SCSI disk).

N

The final number denotes the partition. The first four (primary or extended) partitions are numbered **1** through **4**. Logical partitions start at **5**. So, for example, **/dev/hda3** is the third primary or extended partition on the first IDE hard disk, and **/dev/sdb6** is the second logical partition on the second SCSI hard disk.

Note

There is no part of this naming convention that is based on partition type; unlike DOS/Windows, *all* partitions can be identified under Red Hat Enterprise Linux. Of course, this does not mean that Red Hat Enterprise Linux can access data on every type of partition, but in many cases it is possible to access data on a partition dedicated to another operating system.

Keep this information in mind; it makes things easier to understand when you are setting up the partitions Red Hat Enterprise Linux requires.

26.1.6. Disk Partitions and Other Operating Systems

If your Red Hat Enterprise Linux partitions are sharing a hard disk with partitions used by other operating systems, most of the time you will have no problems. However, there are certain combinations of Linux and other operating systems that require extra care.

26.1.7. Disk Partitions and Mount Points

One area that many people new to Linux find confusing is the matter of how partitions are used and accessed by the Linux operating system. In DOS/Windows, it is relatively simple: Each partition gets a "drive letter." You then use the correct drive letter to refer to files and directories on its corresponding partition.

This is entirely different from how Linux deals with partitions and, for that matter, with disk storage in general. The main difference is that each partition is used to form part of the storage necessary to support a single set of files and directories. This is done by associating a partition with a directory through a process known as *mounting*. Mounting a partition makes its storage available starting at the specified directory (known as a *mount point*).

For example, if partition **/dev/hda5** is mounted on **/usr/**, that would mean that all files and directories under **/usr/** physically reside on **/dev/hda5**. So the file **/usr/share/doc/FAQ/txt/Linux-FAQ** would be stored on **/dev/hda5**, while the file **/etc/gdm/custom.conf** would not.

Continuing our example, it is also possible that one or more directories below **/usr/** would be mount points for other partitions. For instance, a partition (say, **/dev/hda7**) could be mounted on **/usr/local/**, meaning that **/usr/local/man/whatis** would then reside on **/dev/hda7** rather than **/dev/hda5**.

26.1.8. How Many Partitions?

At this point in the process of preparing to install Red Hat Enterprise Linux, you must give some consideration to the number and size of the partitions to be used by your new operating system. The question of "how many partitions" continues to spark debate within the Linux community and, without any end to the debate in sight, it is safe to say that there are probably as many partition layouts as there are people debating the issue.

Keeping this in mind, we recommend that, unless you have a reason for doing otherwise, you should at least create the following partitions: **swap**, **/boot/** (or a **/boot/efi/** partition for Itanium systems), a **/var/** partition for Itanium systems, and **/** (root).

For more information, refer to Section 4.19.4, “Recommended Partitioning Scheme”.

[11] Blocks really *are* consistently sized, unlike our illustrations. Keep in mind, also, that an average disk drive contains thousands of blocks. But for the purposes of this discussion, please ignore these minor discrepancies.

Part V. Basic System Recovery

When things go wrong, there are ways to fix problems. However, these methods require that you understand the system well. This section describes how to boot into rescue mode, single-user mode, and emergency mode, where you can use your own knowledge to repair the system..

Chapter 27. Basic System Recovery

When things go wrong, there are ways to fix problems. However, these methods require that you understand the system well. This chapter describes how to boot into rescue mode, single-user mode, and emergency mode, where you can use your own knowledge to repair the system.

27.1. Common Problems

You might need to boot into one of these recovery modes for any of the following reasons:

- You are unable to boot normally into Red Hat Enterprise Linux (runlevel 3 or 5).
- You are having hardware or software problems, and you want to get a few important files off of your system's hard drive.
- You forgot the root password.

27.1.1. Unable to Boot into Red Hat Enterprise Linux

This problem is often caused by the installation of another operating system after you have installed Red Hat Enterprise Linux. Some other operating systems assume that you have no other operating system(s) on your computer. They overwrite the Master Boot Record (MBR) that originally contained the GRUB boot loader. If the boot loader is overwritten in this manner, you cannot boot Red Hat Enterprise Linux unless you can get into rescue mode and reconfigure the boot loader.

Another common problem occurs when using a partitioning tool to resize a partition or create a new partition from free space after installation, and it changes the order of your partitions. If the partition number of your `/` partition changes, the boot loader might not be able to find it to mount the partition. To fix this problem, boot in rescue mode and modify the **`/boot/grub/grub.conf`** file.

For instructions on how to reinstall the GRUB boot loader from a rescue environment, refer to Section 27.2.1, “Reinstalling the Boot Loader”.

27.1.2. Hardware/Software Problems

This category includes a wide variety of different situations. Two examples include failing hard drives and specifying an invalid root device or kernel in the boot loader configuration file. If either of these occur, you might not be able to reboot into Red Hat Enterprise Linux. However, if you boot into one of the system recovery modes, you might be able to resolve the problem or at least get copies of your most important files.

27.1.3. Root Password

What can you do if you forget your root password? To reset it to a different password, boot into rescue mode or single-user mode, and use the **`passwd`** command to reset the root password.

27.2. Booting into Rescue Mode

Rescue mode provides the ability to boot a small Red Hat Enterprise Linux environment entirely from CD-ROM, or some other boot method, instead of the system's hard drive.

As the name implies, rescue mode is provided to rescue you from something. During normal operation, your Red Hat Enterprise Linux system uses files located on your system's hard drive to do everything — run programs, store your files, and more.

However, there may be times when you are unable to get Red Hat Enterprise Linux running completely enough to access files on your system's hard drive. Using rescue mode, you can access the files stored on your system's hard drive, even if you cannot actually run Red Hat Enterprise Linux from that hard drive.

To boot into rescue mode, you must be able to boot the system using one of the following methods [12]:

- By booting the system from an installation boot CD-ROM.
- By booting the system from other installation boot media, such as USB flash devices.
- By booting the system from the Red Hat Enterprise Linux CD-ROM #1.

Once you have booted using one of the described methods, add the keyword **rescue** as a kernel parameter. For example, for an x86 system, type the following command at the installation boot prompt:

```
linux rescue
```

You are prompted to answer a few basic questions, including which language to use. It also prompts you to select where a valid rescue image is located. Select from **Local CD-ROM**, **Hard Drive**, **NFS image**, **FTP**, or **HTTP**. The location selected must contain a valid installation tree, and the installation tree must be for the same version of Red Hat Enterprise Linux as the Red Hat Enterprise Linux disk from which you booted. If you used a boot CD-ROM or other media to start rescue mode, the installation tree must be from the same tree from which the media was created. For more information about how to setup an installation tree on a hard drive, NFS server, FTP server, or HTTP server, refer to the earlier section of this guide.

If you select a rescue image that does not require a network connection, you are asked whether or not you want to establish a network connection. A network connection is useful if you need to backup files to a different computer or install some RPM packages from a shared network location, for example.

The following message is displayed:

> The rescue environment will now attempt to find your Linux installation and mount it under the directory /mnt/sysimage. You can then make any changes required to your system. If you want to proceed with this step choose 'Continue'. You can also choose to mount your file systems read-only instead of read-write by choosing 'Read-only'. If for some reason this process fails you can choose 'Skip' and this step will be skipped and you will go directly to a command shell.

If you select **Continue**, it attempts to mount your file system under the directory **/mnt/sysimage/**. If it fails to mount a partition, it notifies you. If you select **Read-Only**, it attempts to mount your file system under the directory **/mnt/sysimage/**, but in read-only mode. If you select **Skip**, your file system is not mounted. Choose **Skip** if you think your file system is corrupted.

Once you have your system in rescue mode, a prompt appears on VC (virtual console) 1 and VC 2 (use the **Ctrl**-**Alt**-**F1** key combination to access VC 1 and **Ctrl**-**Alt**-**F2** to access VC 2):

```
sh-3.00b#
```

If you selected **Continue** to mount your partitions automatically and they were mounted successfully, you are in single-user mode.

Even if your file system is mounted, the default root partition while in rescue mode is a temporary root partition, not the root partition of the file system used during normal user mode (runlevel 3 or 5). If you selected to mount your file system and it mounted successfully, you can change the root partition of the rescue mode environment to the root partition of your file system by executing the following command:

```
chroot /mnt/sysimage
```

This is useful if you need to run commands such as **rpm** that require your root partition to be mounted as **/**. To exit the **chroot** environment, type **exit** to return to the prompt.

If you selected **Skip**, you can still try to mount a partition or LVM2 logical volume manually inside rescue mode by creating a directory such as ***/foo***, and typing the following command:

```
mount -t ext3 /dev/mapper/VolGroup00-LogVol02 /foo
```

In the above command, ***/foo*** is a directory that you have created and ***/dev/mapper/VolGroup00-LogVol02*** is the LVM2 logical volume you want to mount. If the partition is of type **ext2**, replace **ext3** with **ext2**.

If you do not know the names of all physical partitions, use the following command to list them:

```
fdisk -l
```

If you do not know the names of all LVM2 physical volumes, volume groups, or logical volumes, use the following commands to list them:

```
pvdisplay
```

```
vgdisplay
```

```
lvdisplay
```

From the prompt, you can run many useful commands, such as:

- **ssh**, **scp**, and **ping** if the network is started
- **dump** and **restore** for users with tape drives
- **parted** and **fdisk** for managing partitions
- **rpm** for installing or upgrading software
- **joe** for editing configuration files

> **Note**
>
> If you try to start other popular editors such as **emacs**, **pico**, or **vi**, the **joe** editor is started.

27.2.1. Reinstalling the Boot Loader

In many cases, the GRUB boot loader can mistakenly be deleted, corrupted, or replaced by other operating systems.

The following steps detail the process on how GRUB is reinstalled on the master boot record:

- Boot the system from an installation boot medium.
- Type **`linux rescue`** at the installation boot prompt to enter the rescue environment.
- Type **`chroot /mnt/sysimage`** to mount the root partition.
- Type **`/usr/sbin/grub-install`** ***`bootpart`*** to reinstall the GRUB boot loader, where *bootpart* is the boot partition (typically, /dev/sda).
- Review the **`/boot/grub/grub.conf`** file, as additional entries may be needed for GRUB to control additional operating systems.
- Reboot the system.

27.3. Booting into Single-User Mode

One of the advantages of single-user mode is that you do not need a boot CD-ROM; however, it does not give you the option to mount the file systems as read-only or not mount them at all.

If your system boots, but does not allow you to log in when it has completed booting, try single-user mode.

In single-user mode, your computer boots to runlevel 1. Your local file systems are mounted, but your network is not activated. You have a usable system maintenance shell. Unlike rescue mode, single-user mode automatically tries to mount your file system. *Do not use single-user mode if your file system cannot be mounted successfully.* You cannot use single-user mode if the runlevel 1 configuration on your system is corrupted.

On an x86 system using GRUB, use the following steps to boot into single-user mode:

1. At the GRUB splash screen at boot time, press any key to enter the GRUB interactive menu.
2. Select **`Red Hat Enterprise Linux`** with the version of the kernel that you wish to boot and type **`a`** to append the line.
3. Go to the end of the line and type **`single`** as a separate word (press the **`Spacebar`** and then type **`single`**). Press **`Enter`** to exit edit mode.

27.4. Booting into Emergency Mode

In emergency mode, you are booted into the most minimal environment possible. The root file system is mounted read-only and almost nothing is set up. The main advantage of emergency mode over single-user mode is that the **`init`** files are not loaded. If **`init`** is corrupted or not working, you can still mount file systems to recover data that could be lost during a re-installation.

To boot into emergency mode, use the same method as described for single-user mode in Section 27.3, “Booting into Single-User Mode” with one exception, replace the keyword **`single`** with the keyword **`emergency`**.

[12] Refer to the earlier sections of this guide for more details.

Chapter 28. Rescue Mode on POWER Systems

You can use the installation disks in rescue mode, in case your system does not boot. Rescue mode gives you access to the disk partitions on your system so you can make any changes necessary to rescue your installation.

To boot in rescue mode on IBM System i with power control NWSD , follow the instructions for booting the installation program, with this change: In the NWSD, set the IPL parameters field to **`'rescue'`** (including the quotes), or to **`'dd rescue'`** if you need to load the SCSI driver. On other systems, specify the **`rescue`** or **`dd rescue`** (without quotes) after the default kernel name at the YABOOT prompt.

After the Language Selection screen (Section 12.13, “Language Selection”), the installation program attempts to mount the disk partitions on your system. It then presents you with a shell prompt where you can make the changes you need. These changes may include storing the kernel and command line into the IPL source, as described in the Installation Complete section (Section 12.26, “Installation Complete”).

When your changes are complete, you can exit the shell using **`exit 0`**. This causes a reboot from the C side. To reboot from the A or B side or from *NWSSTG, you should vary off the system instead of exiting the shell.

28.1. Special Considerations for Accessing the SCSI Utilities from Rescue Mode

If your system uses Native DASD disks, you may need access to the SCSI utilities from rescue mode. These utilities are located on the driver disc CD. The driver disc CD cannot be mounted from rescue mode unless special steps are taken. These steps are described below.

If you have a second CD-ROM drive assigned to your Linux system, you can mount the driver disc CD in the second drive.

If you have only one CD-ROM drive, you must set up an NFS boot, using the following steps:

1. Boot from the CD-ROM with the **`linux rescue askmethod`** command. This allows you to manually select NFS as the source of your rescue media instead of defaulting to the CD-ROM drive.
2. Copy the first installation disc onto a file system of another Linux system.
3. Make this copy of the installation disc available through NFS or FTP.
4. Vary off or power down the system you need to rescue. Set its IPL parameters as instructed for booting the Installation discs in rescue mode, except that the IPL source should point to the copy of **`boot.img`** on your IFS (from step 1, above).
5. *Make sure the installation disc is not in your CD-ROM drive.*
6. IPL the Linux system.
7. Follow the prompts as described in Chapter 28, *Rescue Mode on POWER Systems*. An additional prompt for the installation source appears. Select NFS or FTP (as appropriate) and complete the following network configuration screen.
8. When the Linux system has booted into rescue mode, the CD-ROM drive is available for use and you can mount the driver media to access the SCSI utilities.

Part VI. Advanced Installation and Deployment

The *Red Hat Enterprise Linux Installation Guide* discusses the installation of Red Hat Enterprise Linux and some basic post-installation troubleshooting. However, advanced installation options are also covered in this manual. This part provides instructions for *kickstart* (an automated installation technique)and all related tools. Use this part in conjunction with the first part of the *Red Hat Enterprise Linux Installation Guide* to perform any of these advanced installation tasks.

Chapter 29. Disk Encryption Guide

Note

Red Hat Enterprise Linux 5.3 now contains support during installation for file system encryption. This is not supported for earlier versions of Red Hat Enterprise Linux.

29.1. What is block device encryption?

Block device encryption protects the data on a block device by encrypting it. To access the device's decrypted contents, a user must provide a passphrase or key as authentication. This provides additional security beyond existing OS security mechanisms in that it protects the device's contents even if it has been physically removed from the system.

29.2. Encrypting block devices using dm-crypt/LUKS

Linux Unified Key Setup (LUKS) is a specification for block device encryption. It establishes an on-disk format for the data, as well as a passphrase/key management policy.

LUKS uses the kernel device mapper subsystem via the **`dm-crypt`** module. This arrangement provides a low-level mapping that handles encryption and decryption of the device's data. User-level operations, such as creating and accessing encrypted devices, are accomplished through the use of the **`cryptsetup`** utility.

29.2.1. Overview of LUKS

- What LUKS does:
  - LUKS encrypts entire block devices
    - LUKS is thereby well-suited for protecting the contents of mobile devices such as:
      - Removable storage media
      - Laptop disk drives
  - The underlying contents of the encrypted block device are arbitrary.
    - This makes it useful for encrypting **`swap`** devices.
    - This can also be useful with certain databases that use specially formatted block devices for data storage.
  - LUKS uses the existing device mapper kernel subsystem.
    - This is the same subsystem used by LVM, so it is well tested.
  - LUKS provides passphrase strengthening.
    - This protects against dictionary attacks.
  - LUKS devices contain multiple key slots.
    - This allows users to add backup keys/passphrases.

- What LUKS does *not* do:
  - LUKS is not well-suited for applications requiring many (more than eight) users to have distinct access keys to the same device.
  - LUKS is not well-suited for applications requiring file-level encryption.

For more information on LUKS, refer to http://code.google.com/p/cryptsetup/

29.2.2. How will I access the encrypted devices after installation? (System Startup)

During system startup you will be presented with a passphrase prompt. After the correct passphrase has been provided the system will continue to boot normally. If you used different passphrases for multiple encrypted devices you may need to enter more than one passphrase during the startup.

> **Tip**
>
> Consider using the same passphrase for all encrypted block devices in a given system. This will simplify system startup and you will have fewer passphrases to remember. Just make sure you choose a good passphrase!

29.2.3. Choosing a Good Passphrase

While dm-crypt/LUKS supports both keys and passphrases, the anaconda installer only supports the use of passphrases for creating and accessing encrypted block devices during installation.

LUKS does provide passphrase strengthening but it is still a good idea to choose a good (meaning "difficult to guess") passphrase. Note the use of the term "passphrase", as opposed to the term "password". This is intentional. Providing a phrase containing multiple words to increase the security of your data is important.

29.3. Creating Encrypted Block Devices in Anaconda

You can create encrypted devices during system installation. This allows you to easily configure a system with encrypted partitions.

To enable block device encryption, check the "Encrypt System" checkbox when selecting automatic partitioning or the "Encrypt" checkbox when creating an individual partition, software RAID array, or logical volume. After you finish partitioning, you will be prompted for an encryption passphrase. This passphrase will be required to access the encrypted devices. If you have pre-existing LUKS devices and provided correct passphrases for them earlier in the install process the passphrase entry dialog will also contain a checkbox. Checking this checkbox indicates that you would like the new passphrase to be added to an available slot in each of the pre-existing encrypted block devices.

Important — Global Passphrases Not Supported

Devices encrypted with LUKS can share a global passphrase. When a system contains more than two encrypted block devices, **anaconda** offers you the option to set a global passphrase for them. However, although **anaconda** can set this passphrase correctly, the use of global passphrases is not supported by the init scripts in Red Hat Enterprise Linux 5.

Therefore, even if you set a global passphrase during installation, you must still supply individual passphrases for each encrypted block device every time that the system boots.

Tip

Checking the "Encrypt System" checkbox on the "Automatic Partitioning" screen and then choosing "Create custom layout" does not cause any block devices to be encrypted automatically.

Tip

You can use **`kickstart`** to set a separate passphrase for each new encrypted block device.

29.3.1. What Kinds of Block Devices Can Be Encrypted?

Most types of block devices can be encrypted using LUKS. From anaconda you can encrypt partitions, LVM physical volumes, LVM logical volumes, and software RAID arrays.

29.4. Creating Encrypted Block Devices on the Installed System After Installation

Encrypted block devices can be created and configured after installation.

29.4.1. Create the block devices

Create the block devices you want to encrypt by using **`parted`**, **`pvcreate`**, **`lvcreate`** and **`mdadm`**.

29.4.2. Optional: Fill the device with random data

Filling <device> (eg: **`/dev/sda3`**) with random data before encrypting it greatly increases the strength of the encryption. The downside is that it can take a very long time.

Warning

The commands below will destroy any existing data on the device.

- The best way, which provides high quality random data but takes a long time (several minutes per gigabyte on most systems):

  ```
  dd if=/dev/urandom of=<device>
  ```

- Fastest way, which provides lower quality random data:

  ```
  badblocks -c 10240 -s -w -t random -v <device>
  ```

29.4.3. Format the device as a dm-crypt/LUKS encrypted device

> **Warning**
>
> The command below will destroy any existing data on the device.

```
cryptsetup luksFormat <device>
```

> **Tip**
>
> For more information, read the **`cryptsetup(8)`** man page.

After supplying the passphrase twice the device will be formatted for use. To verify, use the following command:

```
cryptsetup isLuks <device> && echo Success
```

To see a summary of the encryption information for the device, use the following command:

```
cryptsetup luksDump <device>
```

29.4.4. Create a mapping to allow access to the device's decrypted contents

To access the device's decrypted contents, a mapping must be established using the kernel **`device-mapper`**.

It is useful to choose a meaningful name for this mapping. LUKS provides a UUID (Universally Unique Identifier) for each device. This, unlike the device name (eg: **`/dev/sda3`**), is guaranteed to remain constant as long as the LUKS header remains intact. To find a LUKS device's UUID, run the following command:

```
cryptsetup luksUUID <device>
```

An example of a reliable, informative and unique mapping name would be **`luks-<uuid>`**, where <uuid> is replaced with the device's LUKS UUID (eg: **`luks-50ec957a-5b5a-47ee-85e6-f8085bbc97a8`**). This naming convention might seem unwieldy but is it not necessary to type it often.

```
cryptsetup luksOpen <device> <name>
```

There should now be a device node, **/dev/mapper/<name>**, which represents the decrypted device. This block device can be read from and written to like any other unencrypted block device.

To see some information about the mapped device, use the following command:

```
dmsetup info <name>
```

> **Tip**
>
> For more information, read the **dmsetup(8)** man page.

29.4.5. Create filesystems on the mapped device, or continue to build complex storage structures using the mapped device

Use the mapped device node (**/dev/mapper/<name>**) as any other block device. To create an **ext2** filesystem on the mapped device, use the following command:

```
mke2fs /dev/mapper/<name>
```

To mount this filesystem on **/mnt/test**, use the following command:

> **Important**
>
> The directory **/mnt/test** must exist before executing this command.

```
mount /dev/mapper/<name> /mnt/test
```

29.4.6. Add the mapping information to /etc/crypttab

In order for the system to set up a mapping for the device, an entry must be present in the **/etc/crypttab** file. If the file doesn't exist, create it and change the owner and group to root (**root:root**) and change the mode to **0744**. Add a line to the file with the following format:

```
<name>  <device>  none
```

The <device> field should be given in the form "UUID=<luks_uuid>", where <luks_uuid> is the LUKS uuid as given by the command **cryptsetup luksUUID <device>**. This ensures the correct device will be identified and used even if the device node (eg: **/dev/sda5**) changes.

> **Tip**
>
> For details on the format of the **/etc/crypttab** file, read the **crypttab(5)** man page.

29.4.7. Add an entry to /etc/fstab

Add an entry to /etc/fstab. This is only necessary if you want to establish a persistent association between the device and a mountpoint. Use the decrypted device, **/dev/mapper/<name>** in the **/etc/fstab** file.

In many cases it is desirable to list devices in **/etc/fstab** by UUID or by a filesystem label. The main purpose of this is to provide a constant identifier in the event that the device name (eg: **/dev/sda4**) changes. LUKS device names in the form of **/dev/mapper/luks-<luks_uuid>** are based only on the device's LUKS UUID, and are therefore guaranteed to remain constant. This fact makes them suitable for use in **/etc/fstab**.

Title

For details on the format of the **/etc/fstab** file, read the **fstab(5)** man page.

29.5. Common Post-Installation Tasks

The following sections are about common post-installation tasks.

29.5.1. Set a randomly generated key as an additional way to access an encrypted block device

These sections are about generating keys and adding keys.

29.5.1.1. Generate a key

This will generate a 256-bit key in the file **$HOME/keyfile**.

```
dd if=/dev/urandom of=$HOME/keyfile bs=32 count=1
chmod 600 $HOME/keyfile
```

29.5.1.2. Add the key to an available keyslot on the encrypted device

```
cryptsetup luksAddKey <device> ~/keyfile
```

29.5.2. Add a new passphrase to an existing device

```
cryptsetup luksAddKey <device>
```

After being prompted for any one of the existing passphrases for authentication, you will be prompted to enter the new passphrase.

29.5.3. Remove a passphrase or key from a device

```
cryptsetup luksRemoveKey <device>
```

You will be prompted for the passphrase you wish to remove and then for any one of the remaining passphrases for authentication.

Chapter 30. Installing Through VNC

Many enterprise customers perform interactive installations on systems in datacenters. These systems are often, but not always, installed in a rack environment and do not have a display, keyboard, or mouse. Additionally, a lot of these systems even lack the ability to connect a graphical display. Given that enterprise hardware rarely needs that ability at the physical system, this hardware configuration is acceptable.

The Red Hat Enterprise Linux installer (**anaconda**) offers users two interactive modes of operation. The original mode is a text-based interface. The newer mode uses GTK+ and runs in the X Window environment. The purpose of this document is to explain how the graphical installation mode can be used in enterprise environments, even when the system lacks a proper display and input devices typically associated with a workstation.

The primary driver behind the document is to encourage use of the graphical installer, even in enterprise environments. The text mode environment lacks a lot of capabilities found in the graphical mode. Many users still feel that the text mode interface provides them with additional power or configuration ability not found in the graphical version. The opposite is true. Much less development effort is put in to the text mode environment and specific things (e.g., LVM configuration) are deliberately left out of the text mode environment. The reasons for this are:

- Less screen real estate for creating user interfaces similar to those found in the graphical mode.
- Difficult internationalization support.
- Desire to maintain a single interactive installation code path.

All of these reasons and more are why you, the enterprise customer, should be making using of the **Virtual Network Computing** (VNC) mode offered in **anaconda**. VNC allows the graphical mode of the installer to run locally, but display on a system connected to the network.

30.1. VNC Viewer

Performing a VNC installation requires a VNC viewer running on your workstation or other terminal computer. Locations where you might want a VNC viewer installed:

- Your workstation
- Laptop on a datacenter crash cart

VNC is open source software licensed under the GNU General Public License. Versions exist for Linux, Windows, and MacOS X. Here are some recommended VNC viewers:

- **vncviewer** is available on Red Hat Enterprise Linux by installing the *vnc* package:

  ```
  # yum install vnc
  ```

- TightVNC is available for Windows at http://www.tightvnc.com/
- MacOS X includes built-in VNC support as of version 10.5. In the **Finder**, click the **Go** menu and choose **Connect to Server**. In the server address field, you can type **vnc://*SERVER*:*DISPLAY***, where SERVER is the IP address or DNS host name of the VNC server you wish to connect to and DISPLAY is the VNC display number (usually 1), and click **Connect**.

Once you have verified you have a VNC viewer available, it's time to start the installation.

30.2. VNC Modes in Anaconda

30.2. VNC Modes in Anaconda

Anaconda offers two modes for VNC installation. The mode you select will depend on the network configuration in your environment.

30.2.1. Direct Mode

Direct mode VNC in anaconda is when the client initiates a connection to the VNC server running in anaconda. Anaconda will tell you when to initiate this connection in the VNC viewer. Direct mode can be activated by either of the following commands:

- Specify **vnc** as a boot argument.
- Specify the **vnc** command in the kickstart file used for installation.

When you activate VNC mode, anaconda will complete the first stage of the installer and then start VNC to run the graphical installer. The installer will display a message on the console in the following format:

```
Running anaconda VERSION, the PRODUCT system installer - please wait...
```

Anaconda will also tell you the IP address and display number to use in your VNC viewer. At this point, you need to start the VNC viewer and connect to the target system to continue the installation. The VNC viewer will present anaconda to you in graphical mode.

There are some disadvantages to direct mode, including:

- Requires visual access to the system console to see the IP address and port to connect the VNC viewer to.
- Requires interactive access to the system console to complete the first stage of the installer.

If either of these disadvantages would prevent you from using direct mode VNC in anaconda, then connect mode is probably more suited to your environment.

30.2.2. Connect Mode

Certain firewall configurations or instances where the target system is configured to obtain a dynamic IP address may cause trouble with the direct VNC mode in anaconda. In addition, if you lack a console on the target system to see the message that tells you the IP address to connect to, then you will not be able to continue the installation.

The VNC connect mode changes how VNC is started. Rather than anaconda starting up and waiting for you to connect, the VNC connect mode allows anaconda to automatically connect to your view. You won't need to know the IP address of the target system in this case.

To activate the VNC connect mode, pass the **vncconnect** boot parameter:

```
boot: linux vncconnect=HOST
```

Replace HOST with your VNC viewer's IP address or DNS host name. Before starting the installation process on the target system, start up your VNC viewer and have it wait for an incoming connection.

Start the installation and when your VNC viewer displays the graphical installer, you are ready to go.

30.3. Installation Using VNC

Now that you have installed a VNC viewer application and selected a VNC mode for use in anaconda, you are ready to begin the installation.

30.3.1. Installation Example

The easiest way to perform an installation using VNC is to connect another computer directly to the network port on the target system. The laptop on a datacenter crash cart usually fills this role. If you are performing your installation this way, make sure you follow these steps:

1. Connect the laptop or other workstation to the target system using a crossover cable. If you are using regular patch cables, make sure you connect the two systems using a small hub or switch. Most recent Ethernet interfaces will automatically detect if they need to be crossover or not, so it may be possible to connect the two systems directly using a regular patch cable.

2. Configure the VNC viewer system to use a RFC 1918 address with no gateway. This private network connection will only be used for the purpose of installation. Configure the VNC viewer system to be 192.168.100.1/24. If that address is in use, just pick something else in the RFC 1918 address space that is available to you.

3. Start the RHEL installation on the target system.

   a. Booting the installation DVD or CD.

      If booting the installation media (CD or DVD), make sure **vnc** is passed as a boot parameter. To add the **vnc** parameter, you will need a console attached to the target system that allows you to interact with the boot process. Enter the following at the prompt:

      ```
      boot: linux vnc
      ```

   b. Boot over the network.

      If the target system is configured with a static IP address, add the **vnc** command to the kickstart file. If the target system is using DHCP, add **vncconnect=*HOST*** to the boot arguments for the target system. HOST is the IP address or DNS host name of the VNC viewer system. Enter the following at the prompt:

      ```
      boot: linux vncconnect=HOST
      ```

4. When prompted for the network configuration on the target system, assign it an available RFC 1918 address in the same network you used for the VNC viewer system. For example, 192.168.100.2/24.

   > **Note**
   >
   > This IP address is only used during installation. You will have an opportunity to configure the final network settings, if any, later in the installer.

5. Once the installer indicates it is starting anaconda, you will be instructed to connect to the system using the VNC viewer. Connect to the viewer and follow the graphical installation mode instructions found in the product documentation.

30.3.2. Kickstart Considerations

If your target system will be booting over the network, VNC is still available. Just add the **vnc** command to the kickstart file for the system. You will be able to connect to the target system using your VNC viewer and monitor the installation progress. The address to use is the one the system is configured with via the kickstart file.

If you are using DHCP for the target system, the reverse **vncconnect** method may work better for you. Rather than adding the **vnc** boot parameter to the kickstart file, add the **vncconnect=*HOST*** parameter to the list of boot arguments for the target system. For HOST, put the IP address or DNS host name of the VNC viewer system. See the next section for more details on using the vncconnect mode.

30.3.3. Firewall Considerations

If you are performing the installation where the VNC viewer system is a workstation on a different subnet from the target system, you may run in to network routing problems. VNC works fine so long as your viewer system has a route to the target system and ports 5900 and 5901 are open. If your environment has a firewall, make sure ports 5900 and 5901 are open between your workstation and the target system.

In addition to passing the **vnc** boot parameter, you may also want to pass the **vncpassword** parameter in these scenarios. While the password is sent in plain text over the network, it does provide an extra step before a viewer can connect to a system. Once the viewer connects to the target system over VNC, no other connections are permitted. These limitations are usually sufficient for installation purposes.

Important

Be sure to use a temporary password for the **vncpassword** option. It should not be a password you use on any systems, especially a real root password.

If you continue to have trouble, consider using the **vncconnect** parameter. In this mode of operation, you start the viewer on your system first telling it to listen for an incoming connection. Pass **vncconnect=*HOST*** at the boot prompt and the installer will attempt to connect to the specified HOST (either a hostname or IP address).

30.4. References

- VNC description at Wikipedia: http://en.wikipedia.org/wiki/Vnc
- TightVNC: http://www.tightvnc.com/
- RFC 1918 - Address Allocation for Private Networks: http://www.ietf.org/rfc/rfc1918.txt
- Anaconda boot options: http://fedoraproject.org/wiki/Anaconda/Options
- Kickstart documentation: http://fedoraproject.org/wiki/Anaconda/Kickstart

Chapter 31. Kickstart Installations

31.1. What are Kickstart Installations?

Many system administrators would prefer to use an automated installation method to install Red Hat Enterprise Linux on their machines. To answer this need, Red Hat created the kickstart installation method. Using kickstart, a system administrator can create a single file containing the answers to all the questions that would normally be asked during a typical installation.

Kickstart files can be kept on a single server system and read by individual computers during the installation. This installation method can support the use of a single kickstart file to install Red Hat Enterprise Linux on multiple machines, making it ideal for network and system administrators.

Kickstart provides a way for users to automate a Red Hat Enterprise Linux installation.

31.2. How Do You Perform a Kickstart Installation?

Kickstart installations can be performed using a local CD-ROM, a local hard drive, or via NFS, FTP, or HTTP.

To use kickstart, you must:

1. Create a kickstart file.
2. Create a boot media with the kickstart file or make the kickstart file available on the network.
3. Make the installation tree available.
4. Start the kickstart installation.

This chapter explains these steps in detail.

31.3. Creating the Kickstart File

The kickstart file is a simple text file, containing a list of items, each identified by a keyword. You can create it by using the **Kickstart Configurator** application, or writing it from scratch. The Red Hat Enterprise Linux installation program also creates a sample kickstart file based on the options that you selected during installation. It is written to the file **`/root/anaconda-ks.cfg`**. You should be able to edit it with any text editor or word processor that can save files as ASCII text.

First, be aware of the following issues when you are creating your kickstart file:

- Sections must be specified *in order*. Items within the sections do not have to be in a specific order unless otherwise specified. The section order is:
  - Command section — Refer to Section 31.4, “Kickstart Options” for a list of kickstart options. You must include the required options.
  - The **`%packages`** section — Refer to Section 31.5, “Package Selection” for details.
  - The **`%pre`** and **`%post`** sections — These two sections can be in any order and are not required. Refer to Section 31.6, “Pre-installation Script” and Section 31.7, “Post-installation Script” for details.
- Items that are not required can be omitted.

- Omitting any required item results in the installation program prompting the user for an answer to the related item, just as the user would be prompted during a typical installation. Once the answer is given, the installation continues unattended (unless it finds another missing item).
- Lines starting with a pound (also known as hash) sign (#) are treated as comments and are ignored.
- For kickstart *upgrades*, the following items are required:
  - Language
  - Installation method
  - Device specification (if device is needed to perform the installation)
  - Keyboard setup
  - The **upgrade** keyword
  - Boot loader configuration

  If any other items are specified for an upgrade, those items are ignored (note that this includes package selection).

31.4. Kickstart Options

The following options can be placed in a kickstart file. If you prefer to use a graphical interface for creating your kickstart file, use the **Kickstart Configurator** application. Refer to Chapter 32, *Kickstart Configurator* for details.

> **Note**
>
> If the option is followed by an equals mark (=), a value must be specified after it. In the example commands, options in brackets ([]) are optional arguments for the command.

autopart (optional)

Automatically create partitions — 1 GB or more root (/) partition, a swap partition, and an appropriate boot partition for the architecture. One or more of the default partition sizes can be redefined with the **part** directive.

- **--encrypted** — Should all devices with support be encrypted by default? This is equivalent to checking the **Encrypt** checkbox on the initial partitioning screen.
- **--passphrase=** — Provide a default system-wide passphrase for all encrypted devices.

ignoredisk (optional)

Causes the installer to ignore the specified disks. If the target system contains read-only devices, they must be specified with **ignoredisk** or the installation will stop and prompt for whether it should ignore the device.

ignoredisk is useful if you use autopartition and want to be sure that some disks are ignored. For example, without **ignoredisk**, attempting to deploy on a SAN-cluster the kickstart would fail, as the installer detects passive paths to the SAN that return no partition

table. The **--only-use** option specifies that only the disks listed will be used during installation.

The **ignoredisk** option is also useful if you have multiple paths to your disks.

The syntax is:

```
ignoredisk --drives=drive1,drive2,...
```

where *driveN* is one of **sda**, **sdb**,..., **hda**,... etc.

- **--only-use** — specifies a list of disks for the installer to use. All other disks are ignored. For example, to use disk **sda** during installation and ignore all other disks:

  ```
  ignoredisk --only-use=sda
  ```

autostep (optional)

Similar to **interactive** except it goes to the next screen for you. It is used mostly for debugging and should not be used when deploying a system because it may disrupt package installation.

- **--autoscreenshot** — Take a screenshot at every step during installation and copy the images over to **/root/anaconda-screenshots** after installation is complete. This is most useful for documentation.

auth or authconfig (required)

Sets up the authentication options for the system. It is similar to the **authconfig** command, which can be run after the install. By default, passwords are normally encrypted and are not shadowed.

- **--enablemd5** — Use md5 encryption for user passwords.
- **--enablenis** — Turns on NIS support. By default, **--enablenis** uses whatever domain it finds on the network. A domain should almost always be set by hand with the **--nisdomain=** option.
- **--nisdomain=** — NIS domain name to use for NIS services.
- **--nisserver=** — Server to use for NIS services (broadcasts by default).
- **--useshadow** or **--enableshadow** — Use shadow passwords.
- **--enableldap** — Turns on LDAP support in **/etc/nsswitch.conf**, allowing your system to retrieve information about users (UIDs, home directories, shells, etc.) from an LDAP directory. To use this option, you must install the **nss_ldap** package. You must also specify a server and a base DN (distinguished name) with **--ldapserver=** and **--ldapbasedn=**.
- **--enableldapauth** — Use LDAP as an authentication method. This enables the **pam_ldap** module for authentication and changing passwords, using an LDAP directory. To use this option, you must have the **nss_ldap** package installed. You must also specify a server and a base DN with **--ldapserver=** and **--ldapbasedn=**.

- **--ldapserver=** — If you specified either **--enableldap** or **--enableldapauth**, use this option to specify the name of the LDAP server to use. This option is set in the **/etc/ldap.conf** file.
- **--ldapbasedn=** — If you specified either **--enableldap** or **--enableldapauth**, use this option to specify the DN in your LDAP directory tree under which user information is stored. This option is set in the **/etc/ldap.conf** file.
- **--enableldaptls** — Use TLS (Transport Layer Security) lookups. This option allows LDAP to send encrypted usernames and passwords to an LDAP server before authentication.
- **--enablekrb5** — Use Kerberos 5 for authenticating users. Kerberos itself does not know about home directories, UIDs, or shells. If you enable Kerberos, you must make users' accounts known to this workstation by enabling LDAP, NIS, or Hesiod or by using the **/usr/sbin/useradd** command. If you use this option, you must have the **pam_krb5** package installed.
- **--krb5realm=** — The Kerberos 5 realm to which your workstation belongs.
- **--krb5kdc=** — The KDC (or KDCs) that serve requests for the realm. If you have multiple KDCs in your realm, separate their names with commas (,).
- **--krb5adminserver=** — The KDC in your realm that is also running kadmind. This server handles password changing and other administrative requests. This server must be run on the master KDC if you have more than one KDC.
- **--enablehesiod** — Enable Hesiod support for looking up user home directories, UIDs, and shells. More information on setting up and using Hesiod on your network is in **/usr/share/doc/glibc-2.x.x/README.hesiod**, which is included in the **glibc** package. Hesiod is an extension of DNS that uses DNS records to store information about users, groups, and various other items.
- **--hesiodlhs** — The Hesiod LHS ("left-hand side") option, set in **/etc/hesiod.conf**. This option is used by the Hesiod library to determine the name to search DNS for when looking up information, similar to LDAP's use of a base DN.
- **--hesiodrhs** — The Hesiod RHS ("right-hand side") option, set in **/etc/hesiod.conf**. This option is used by the Hesiod library to determine the name to search DNS for when looking up information, similar to LDAP's use of a base DN.

> **Note**
>
> To look up user information for "jim", the Hesiod library looks up *jim.passwd<LHS><RHS>*, which should resolve to a TXT record that looks like what his passwd entry would look like (**jim:*:501:501:Jungle Jim:/home/jim:/bin/bash**). For groups, the situation is identical, except *jim.group<LHS><RHS>* would be used.
>
> Looking up users and groups by number is handled by making "501.uid" a CNAME for "jim.passwd", and "501.gid" a CNAME for "jim.group". Note that the library does not place a period **.** in front of the LHS and RHS values when performing a search. Therefore the LHS and RHS values need to have a period placed in front of them in order if they require this.

- **--enablesmbauth** — Enables authentication of users against an SMB server (typically a Samba or Windows server). SMB authentication support does not know about home directories, UIDs, or shells. If you enable SMB, you must make users' accounts known to the workstation by enabling LDAP, NIS, or Hesiod or by using the **/usr/sbin/useradd** command to make their accounts known to the workstation. To use this option, you must have the **pam_smb** package installed.

- **--smbservers=** — The name of the server(s) to use for SMB authentication. To specify more than one server, separate the names with commas (,).

- **--smbworkgroup=** — The name of the workgroup for the SMB servers.

- **--enablecache** — Enables the **nscd** service. The **nscd** service caches information about users, groups, and various other types of information. Caching is especially helpful if you choose to distribute information about users and groups over your network using NIS, LDAP, or hesiod.

- **--passalgo** — Enables SHA256 or SHA512 hashing for passphrases. Use **--passalgo=sha256** or **--passalgo=sha512** and remove the **--enablemd5** if present.

bootloader (required)

Specifies how the boot loader should be installed. This option is required for both installations and upgrades.

- **--append=** — Specifies kernel parameters. To specify multiple parameters, separate them with spaces. For example:

```
bootloader --location=mbr --append="hdd=ide-scsi ide=nodma"
```

- **--driveorder** — Specify which drive is first in the BIOS boot order. For example:

```
bootloader --driveorder=sda,hda
```

- **--hvargs** — If using **GRUB**, specifies Xen hypervisor arguments. To specify multiple parameters, separate them with spaces. For example:

```
bootloader --hvargs="dom0_mem=2G dom0_max_vcpus=4"
```

- **--location=** — Specifies where the boot record is written. Valid values are the following: **mbr** (the default), **partition** (installs the boot loader on the first sector of the partition containing the kernel), or **none** (do not install the boot loader).

- **--password=** — If using GRUB, sets the GRUB boot loader password to the one specified with this option. This should be used to restrict access to the GRUB shell, where arbitrary kernel options can be passed.

- **--md5pass=** — If using GRUB, similar to **--password=** except the password should already be encrypted.

- **--upgrade** — Upgrade the existing boot loader configuration, preserving the old entries. This option is only available for upgrades.

clearpart (optional)

Removes partitions from the system, prior to creation of new partitions. By default, no partitions are removed.

> **Note**
>
> If the **`clearpart`** command is used, then the **`--onpart`** command cannot be used on a logical partition.

> **Important — unformatted DASDs on System z**
>
> Red Hat Enterprise Linux 5 cannot use unformatted DASDs when installing with kickstart and the cmdline user interface. Use the following command in your kickstart file and explicitly list all DASDs that you want to be low-level formatted with **dasdfmt** automatically in case they are not yet low-level formatted:
>
> ```
> clearpart --initlabel --drives=names_of_DASDs
> ```
>
> For example:
>
> ```
> clearpart --initlabel --drives=dasda,dasdb,dasdc
> ```

- **`--all`** — Erases all partitions from the system.
- **`--drives=`** — Specifies which drives to clear partitions from. For example, the following clears all the partitions on the first two drives on the primary IDE controller:

  ```
  clearpart --drives=hda,hdb --all
  ```

- **`--initlabel`** — Initializes the disk label to the default for your architecture (for example **`msdos`** for x86 and **`gpt`** for Itanium). It is useful so that the installation program does not ask if it should initialize the disk label if installing to a brand new hard drive.
- **`--linux`** — Erases all Linux partitions.
- **`--none`** (default) — Do not remove any partitions.

`cmdline` **(optional)**

Perform the installation in a completely non-interactive command line mode. Any prompts for interaction halts the install. This mode is useful on IBM System z systems with the x3270 console.

`device` **(optional)**

On most PCI systems, the installation program autoprobes for Ethernet and SCSI cards properly. On older systems and some PCI systems, however, kickstart needs a hint to find the proper devices. The **`device`** command, which tells the installation program to install extra modules, is in this format:

```
device <type> <moduleName> --opts=<options>
```

- *<type>* — Replace with either **`scsi`** or **`eth`**.
- *<moduleName>* — Replace with the name of the kernel module which should be installed.

- **--opts=** — Mount options to use for mounting the NFS export. Any options that can be specified in **/etc/fstab** for an NFS mount are allowed. The options are listed in the **nfs(5)** man page. Multiple options are separated with a comma.

driverdisk (optional)

Driver diskettes can be used during kickstart installations. You must copy the driver diskettes's contents to the root directory of a partition on the system's hard drive. Then you must use the **driverdisk** command to tell the installation program where to look for the driver disk.

```
driverdisk <partition> [--type=<fstype>]
```

Alternatively, a network location can be specified for the driver diskette:

```
driverdisk --source=ftp://path/to/dd.img
driverdisk --source=http://path/to/dd.img
driverdisk --source=nfs:host:/path/to/img
```

- *<partition>* — Partition containing the driver disk.
- **--type=** — File system type (for example, vfat or ext2).

firewall (optional)

This option corresponds to the **Firewall Configuration** screen in the installation program:

```
firewall --enabled|--disabled [--trust=] <device> [--port=]
```

- **--enabled** or **--enable** — Reject incoming connections that are not in response to outbound requests, such as DNS replies or DHCP requests. If access to services running on this machine is needed, you can choose to allow specific services through the firewall.
- **--disabled** or **--disable** — Do not configure any iptables rules.
- **--trust=** — Listing a device here, such as eth0, allows all traffic coming to and from that device to go through the firewall. To list more than one device, use **--trust eth0 --trust eth1**. Do NOT use a comma-separated format such as **--trust eth0, eth1**.
- *<incoming>* — Replace with one or more of the following to allow the specified services through the firewall.
  - **--ssh**
  - **--telnet**
  - **--smtp**
  - **--http**
  - **--ftp**
- **--port=** — You can specify that ports be allowed through the firewall using the port:protocol format. For example, to allow IMAP access through your firewall, specify **imap:tcp**. Numeric ports can also be specified explicitly; for example, to allow UDP

packets on port 1234 through, specify **1234:udp**. To specify multiple ports, separate them by commas.

firstboot (optional)

Determine whether the **Setup Agent** starts the first time the system is booted. If enabled, the **firstboot** package must be installed. If not specified, this option is disabled by default.

- **--enable** or **--enabled** — The **Setup Agent** is started the first time the system boots.
- **--disable** or **--disabled** — The **Setup Agent** is not started the first time the system boots.
- **--reconfig** — Enable the **Setup Agent** to start at boot time in reconfiguration mode. This mode enables the language, mouse, keyboard, root password, security level, time zone, and networking configuration options in addition to the default ones.

halt (optional)

Halt the system after the installation has successfully completed. This is similar to a manual installation, where anaconda displays a message and waits for the user to press a key before rebooting. During a kickstart installation, if no completion method is specified, this option is used as the default.

The **halt** option is roughly equivalent to the **shutdown -h** command.

For other completion methods, refer to the **poweroff**, **reboot**, and **shutdown** kickstart options.

graphical (optional)

Perform the kickstart installation in graphical mode. This is the default.

install (optional)

Tells the system to install a fresh system rather than upgrade an existing system. This is the default mode. For installation, you must specify the type of installation from **cdrom**, **harddrive**, **nfs**, or **url** (for FTP or HTTP installations). The **install** command and the installation method command must be on separate lines.

- **cdrom** — Install from the first CD-ROM drive on the system.
- **harddrive** — Install from a Red Hat installation tree on a local drive, which must be either vfat or ext2.
  - **--biospart=**

    BIOS partition to install from (such as 82).
  - **--partition=**

    Partition to install from (such as sdb2).
  - **--dir=**

    Directory containing the ***variant*** directory of the installation tree.

  For example:

```
harddrive --partition=hdb2 --dir=/tmp/install-tree
```

- **nfs** — Install from the NFS server specified.
  - **--server=**

    Server from which to install (hostname or IP).
  - **--dir=**

    Directory containing the ***variant*** directory of the installation tree.
  - **--opts=**

    Mount options to use for mounting the NFS export. (optional)

  For example:

  ```
  nfs --server=nfsserver.example.com --dir=/tmp/install-tree
  ```

- **url** — Install from an installation tree on a remote server via FTP or HTTP.

  For example:

  ```
  url --url http://<server>/<dir>
  ```

  or:

  ```
  url --url ftp://<username>:<password>@<server>/<dir>
  ```

interactive (optional)

Uses the information provided in the kickstart file during the installation, but allow for inspection and modification of the values given. You are presented with each screen of the installation program with the values from the kickstart file. Either accept the values by clicking **Next** or change the values and click **Next** to continue. Refer to the **autostep** command.

iscsi (optional)

iscsi --ipaddr= [options].

Specifies additional iSCSI storage to be attached during installation. If you use the ***iscsi*** parameter, you must also assign a name to the iSCSI node, using the ***iscsiname*** parameter. The ***iscsiname*** parameter must appear before the ***iscsi*** parameter in the kickstart file.

We recommend that wherever possible you configure iSCSI storage in the system BIOS or firmware (iBFT for Intel systems) rather than use the ***iscsi*** parameter. **Anaconda** automatically detects and uses disks configured in BIOS or firmware and no special configuration is necessary in the kickstart file.

If you must use the ***iscsi*** parameter, ensure that networking is activated at the beginning of the installation, and that the ***iscsi*** parameter appears in the kickstart file before you refer to iSCSI disks with parameters such as ***clearpart*** or ***ignoredisk***.

- **--port=** (mandatory) — the port number (typically, **--port=3260**)
- **--user=** — the username required to authenticate with the target

- **--password=** — the password that corresponds with the username specified for the target
- **--reverse-user=** — the username required to authenticate with the initiator from a target that uses reverse CHAP authentication
- **--reverse-password=** — the password that corresponds with the username specified for the initiator

iscsiname (optional)

Assigns a name to an iSCSI node specified by the iscsi parameter. If you use the ***iscsi*** parameter in your kickstart file, this parameter is mandatory, and you must specify ***iscsiname*** in the kickstart file before you specify ***iscsi***.

key (optional)

Specify an installation key, which is needed to aid in package selection and identify your system for support purposes.

- **--skip** — Skip entering a key. Usually if the key command is not given, anaconda will pause at this step to prompt for a key. This option allows automated installation to continue if you do not have a key or do not want to provide one.

keyboard (required)

Sets system keyboard type. Here is the list of available keyboards on i386, Itanium, and Alpha machines:

```
be-latin1, bg, br-abnt2, cf, cz-lat2, cz-us-qwertz, de, de-latin1,
de-latin1-nodeadkeys, dk, dk-latin1, dvorak, es, et, fi, fi-latin1,
fr, fr-latin0, fr-latin1, fr-pc, fr_CH, fr_CH-latin1, gr, hu, hu101,
is-latin1, it, it-ibm, it2, jp106, la-latin1, mk-utf, no, no-latin1,
pl, pt-latin1, ro_win, ru, ru-cp1251, ru-ms, ru1, ru2,  ru_win,
se-latin1, sg, sg-latin1, sk-qwerty, slovene, speakup,  speakup-lt,
sv-latin1, sg, sg-latin1, sk-querty, slovene, trq, ua,  uk, us, us-acentos
```

The file **/usr/lib/python2.2/site-packages/rhpl/keyboard_models.py** also contains this list and is part of the **rhpl** package.

lang (required)

Sets the language to use during installation and the default language to use on the installed system. For example, to set the language to English, the kickstart file should contain the following line:

```
lang en_US
```

The file **/usr/share/system-config-language/locale-list** provides a list of the valid language codes in the first column of each line and is part of the **system-config-language** package.

Certain languages (mainly Chinese, Japanese, Korean, and Indic languages) are not supported during text mode installation. If one of these languages is specified using the lang command, installation will continue in English though the running system will have the specified language by default.

langsupport (deprecated)

The langsupport keyword is deprecated and its use will cause an error message to be printed to the screen and installation to halt. Instead of using the langsupport keyword, you should now list the support package groups for all languages you want supported in the **%packages** section of your kickstart file. For instance, adding support for French means you should add the following to **%packages**:

```
@french-support
```

logvol (optional)

Create a logical volume for Logical Volume Management (LVM) with the syntax:

```
logvol <mntpoint> --vgname=<name> --size=<size> --name=<name> <options>
```

The options are as follows:

- **--noformat** — Use an existing logical volume and do not format it.
- **--useexisting** — Use an existing logical volume and reformat it.
- **--fstype=** — Sets the file system type for the logical volume. Valid values are **xfs**, **ext2**, **ext3**, **ext4**, **swap**, **vfat**, and **hfs**.
- **--fsoptions=** — Specifies a free form string of options to be used when mounting the filesystem. This string will be copied into the **/etc/fstab** file of the installed system and should be enclosed in quotes.
- **--bytes-per-inode=** — Specifies the size of inodes on the filesystem to be made on the logical volume. Not all filesystems support this option, so it is silently ignored for those cases.
- **--size=** — The minimum size of the logical volume in megabytes. Specify an integer value here, and do not append the number with MB. If the logical volume is set to grow, the minimum size must be provided.
- **--grow=** — Tells the logical volume to grow to fill available space (if any), or up to the maximum size setting, while conforming to other limitations.
- **--maxsize=** — The maximum size in megabytes when the logical volume is set to grow. Specify an integer value here, and do not append the number with MB.
- **--recommended=** — Determine the size of the logical volume automatically.
- **--percent=** — Specify the size of the logical volume as a percentage of available space in the volume group.

Create the partition first, create the logical volume group, and then create the logical volume. For example:

```
part pv.01 --size 3000
volgroup myvg pv.01
logvol / --vgname=myvg --size=2000 --name=rootvol
```

logging (optional)

This command controls the error logging of anaconda during installation. It has no effect on the installed system.

- **--host=** — Send logging information to the given remote host, which must be running a syslogd process configured to accept remote logging.
- **--port=** — If the remote syslogd process uses a port other than the default, it may be specified with this option.
- **--level=** — One of debug, info, warning, error, or critical.

  Specify the minimum level of messages that appear on tty3. All messages will still be sent to the log file regardless of this level, however.

mediacheck (optional)

If given, this will force anaconda to run mediacheck on the installation media. This command requires that installs be attended, so it is disabled by default.

monitor (optional)

If the monitor command is not given, anaconda will use X to automatically detect your monitor settings. Please try this before manually configuring your monitor.

- **--hsync=** — Specifies the horizontal sync frequency of the monitor.
- **--monitor=** — Use specified monitor; monitor name should be from the list of monitors in /usr/share/hwdata/MonitorsDB from the hwdata package. The list of monitors can also be found on the X Configuration screen of the Kickstart Configurator. This is ignored if --hsync or --vsync is provided. If no monitor information is provided, the installation program tries to probe for it automatically.
- **--noprobe=** — Do not try to probe the monitor.
- **--vsync=** — Specifies the vertical sync frequency of the monitor.

mouse (deprecated)

The mouse keyword is deprecated.

multipath (optional)

Specifies a multipath device in the format:

```
multipath --name=mpathX --device=device_name --rule=policy
```

For example:

```
multipath --name=mpath0 --device=/dev/sdc --rule=failover
```

The available options are:

- **--name=** — the name for the multipath device, in the format **mpath*X***, where *X* is an integer.
- **--device=** — the block device connected as a multipath device.
- **--rule=** — a multipath *policy*: **failover**, **multibus**, **group_by_serial**, **group_by_prio**, or **group_by_node_name**. Refer to the multipath manpage for a description of these policies.

network (optional)

Configures network information for the system. If the kickstart installation requires networking (when the Kickstart file is accessed over **HTTP**, **FTP** or **NFS**), the device specified in the first network command is activated with configuration specified by the command. If the **--device=** option is not specified and multiple network devices are available, the device used to access the Kickstart file over the network is selected, or the user is asked to choose the device.

Note that if network configuration is not specified in the first **network** command (for example, if the **--bootproto=** option is missing), the device will be activated with configuration set by boot options. On the installed system, this device will be configured using the default value, which is **--bootproto=dhcp**.

- **--bootproto=** — One of **dhcp**, **bootp**, **static** or **query**.

  The default option is **dhcp**. **bootp** and **dhcp** are treated the same.

  The DHCP method uses a DHCP server system to obtain its networking configuration. As you might guess, the BOOTP method is similar, requiring a BOOTP server to supply the networking configuration. To direct a system to use DHCP:

  ```
  network --bootproto=dhcp
  ```

  To direct a machine to use BOOTP to obtain its networking configuration, use the following line in the kickstart file:

  ```
  network --bootproto=bootp
  ```

  The static method requires that you enter all the required networking information in the kickstart file. As the name implies, this information is static and is used during and after the installation. The line for static networking is more complex, as you must include all network configuration information on one line. You must specify the IP address, netmask, gateway, and nameserver.

  Note that although the presentation of this example on this page has broken the line, in a real kickstart file, you must include all this information on a single line with no break.

  ```
  network --bootproto=static --ip=10.0.2.15 --netmask=255.255.255.0
   --gateway=10.0.2.254 --nameserver=10.0.2.1
  ```

  If you use the static method, be aware of the following two restrictions:

  - All static networking configuration information must be specified on *one* line; you cannot wrap lines using a backslash, for example.
  - You can also configure multiple nameservers here. To do so, specify them as a comma-delimited list in the command line.

    Note that although the presentation of this example on this page has broken the line, in a real kickstart file, you must include all this information on a single line with no break.

    ```
    network --bootproto=static --ip=10.0.2.15 --netmask=255.255.255.0
     --gateway=10.0.2.254 --nameserver 192.168.2.1,192.168.3.1
    ```

  If you set this option to **query**, you will be prompted to manually configure the network options during the actual installation process.

```
network --bootproto=query
```

- **--device=** — Used to specify the network device being configured. For example:

  ```
  network --bootproto=dhcp --device=eth0
  ```

  The above example configures device **eth0** for DHCP.

- **--ip=** — IP address for the machine to be installed.
- **--gateway=** — Default gateway as an IP address.
- **--nameserver=** — Primary nameserver, as an IP address.
- **--nodns** — Do not configure any DNS server.
- **--netmask=** — Netmask for the installed system.
- **--hostname=** — Hostname for the installed system.
- **--ethtool=** — Specifies additional low-level settings for the network device which will be passed to the ethtool program. When **autoneg** is not specified, **autoneg off** is inserted automatically.
- **--essid=** — The network ID for wireless networks.
- **--wepkey=** — The encryption key for wireless networks.
- **--onboot=** — Whether or not to enable the device at boot time.
- **--dhcpclass=** — The DHCP class.
- **--mtu=** — The MTU of the device.
- **--noipv4** — Disable IPv4 on this device.
- **--noipv6** — Disable IPv6 on this device.

part or partition (required for installs, ignored for upgrades)

Creates a partition on the system.

If more than one Red Hat Enterprise Linux installation exists on the system on different partitions, the installation program prompts the user and asks which installation to upgrade.

> **Warning**
>
> All partitions created are formatted as part of the installation process unless **--noformat** and **--onpart** are used.

For a detailed example of **part** in action, refer to Section 31.4.1, “Advanced Partitioning Example”.

- *<mntpoint>* — The *<mntpoint>* is where the partition is mounted and must be of one of the following forms:

  - ***/<path>***

    For example, **/**, **/usr**, **/home**

  - **swap**

    The partition is used as swap space.

    To determine the size of the swap partition automatically, use the **--recommended** option:

    ```
    swap --recommended
    ```

    The recommended maximum swap size for machines with less than 2GB of RAM is twice the amount of RAM. For machines with 2GB or more, this recommendation changes to 2GB plus the amount of RAM.

  - **raid.*<id>***

    The partition is used for software RAID (refer to **raid**).

  - **pv.*<id>***

    The partition is used for LVM (refer to **logvol**).

> **Note**
>
> You can assign any value to the ***<id>*** field, but ensure these values are consistent across volumes and volume groups. The default value for the first volume is 01.

- **--size=** — The minimum partition size in megabytes. Specify an integer value here such as 500. Do not append the number with MB.

- **--grow** — Tells the partition to grow to fill available space (if any), or up to the maximum size setting.

> **Note**
>
> If you use **--grow=** without setting **--maxsize=** on a swap partition, **Anaconda** will limit the maximum size of the swap partition. For systems that have less than 2GB of physical memory, the imposed limit is twice the amount of physical memory. For systems with more than 2GB, the imposed limit is the size of physical memory plus 2GB.

- **--maxsize=** — The maximum partition size in megabytes when the partition is set to grow. Specify an integer value here, and do not append the number with MB.

- **--noformat** — Tells the installation program not to format the partition, for use with the **--onpart** command.

- **--onpart=** or **--usepart=** — Put the partition on the *already existing* device. For example:

```
partition /home --onpart=hda1
```

puts **/home** on **/dev/hda1**, which must already exist.

- **--ondisk=** or **--ondrive=** — Forces the partition to be created on a particular disk. For example, **--ondisk=sdb** puts the partition on the second SCSI disk on the system.
- **--asprimary** — Forces automatic allocation of the partition as a primary partition, or the partitioning fails.
- **--type=** (replaced by **fstype**) — This option is no longer available. Use **fstype**.
- **--fstype=** — Sets the file system type for the partition. Valid values are **xfs**, **ext2**, **ext3**, **ext4**, **swap**, **vfat**, and **hfs**.
- **--start=** — Specifies the starting cylinder for the partition. It requires that a drive be specified with **--ondisk=** or **ondrive=**. It also requires that the ending cylinder be specified with **--end=** or the partition size be specified with **--size=**.
- **--end=** — Specifies the ending cylinder for the partition. It requires that the starting cylinder be specified with **--start=**.
- **--bytes-per-inode=** — Specifies the size of inodes on the filesystem to be made on the partition. Not all filesystems support this option, so it is silently ignored for those cases.
- **--recommended** — Determine the size of the partition automatically.
- **--onbiosdisk** — Forces the partition to be created on a particular disk as discovered by the BIOS.
- **--encrypted** — Specifies that this partition should be encrypted.
- **--passphrase=** — Specifies the passphrase to use when encrypting this partition. Without the above **--encrypted** option, this option does nothing. If no passphrase is specified, the default system-wide one is used, or the installer will stop and prompt if there is no default.
- **--fsoptions=** — Specifies a free form string of options to be used when mounting the filesystem. This string will be copied into the **/etc/fstab** file of the installed system and should be enclosed in quotes.
- **--label=** — assign a label to an individual partition.

Note

If partitioning fails for any reason, diagnostic messages appear on virtual console 3.

poweroff (optional)

Shut down and power off the system after the installation has successfully completed. Normally during a manual installation, anaconda displays a message and waits for the user to press a key before rebooting. During a kickstart installation, if no completion method is specified, the **halt** option is used as default.

The **poweroff** option is roughly equivalent to the **shutdown -p** command.

> **Note**
>
> The **poweroff** option is highly dependent on the system hardware in use. Specifically, certain hardware components such as the BIOS, APM (advanced power management), and ACPI (advanced configuration and power interface) must be able to interact with the system kernel. Contact your manufacturer for more information on you system's APM/ACPI abilities.

For other completion methods, refer to the **halt**, **reboot**, and **shutdown** kickstart options.

raid (optional)

Assembles a software RAID device. This command is of the form:

```
raid <mntpoint> --level=<level> --device=<mddevice> <partitions*>
```

- *<mntpoint>* — Location where the RAID file system is mounted. If it is **/**, the RAID level must be 1 unless a boot partition (**/boot**) is present. If a boot partition is present, the **/boot** partition must be level 1 and the root (**/**) partition can be any of the available types. The *<partitions*>* (which denotes that multiple partitions can be listed) lists the RAID identifiers to add to the RAID array.
- **--level=** — RAID level to use (0, 1, 4, 5, 6, or 10).
- **--device=** — Name of the RAID device to use (such as md0 or md1). RAID devices range from md0 to md15, and each may only be used once.
- **--bytes-per-inode=** — Specifies the size of inodes on the filesystem to be made on the RAID device. Not all filesystems support this option, so it is silently ignored for those cases.
- **--spares=** — Specifies the number of spare drives allocated for the RAID array. Spare drives are used to rebuild the array in case of drive failure.
- **--fstype=** — Sets the file system type for the RAID array. Valid values are **xfs**, **ext2**, **ext3**, **ext4**, **swap**, **vfat**, and **hfs**.
- **--fsoptions=** — Specifies a free form string of options to be used when mounting the filesystem. This string will be copied into the /etc/fstab file of the installed system and should be enclosed in quotes.
- **--noformat** — Use an existing RAID device and do not format the RAID array.
- **--useexisting** — Use an existing RAID device and reformat it.
- **--encrypted** — Specifies that this RAID device should be encrypted.
- **--passphrase=** — Specifies the passphrase to use when encrypting this RAID device. Without the above **--encrypted** option, this option does nothing. If no passphrase is specified, the default system-wide one is used, or the installer will stop and prompt if there is no default.

The following example shows how to create a RAID level 1 partition for **/**, and a RAID level 5 for **/usr**, assuming there are three SCSI disks on the system. It also creates three swap partitions, one on each drive.

```
part raid.01 --size=60 --ondisk=sda
part raid.02 --size=60 --ondisk=sdb
part raid.03 --size=60 --ondisk=sdc
```

```
part swap --size=128 --ondisk=sda
part swap --size=128 --ondisk=sdb
part swap --size=128 --ondisk=sdc
```

```
part raid.11 --size=1 --grow --ondisk=sda
part raid.12 --size=1 --grow --ondisk=sdb
part raid.13 --size=1 --grow --ondisk=sdc
```

```
raid / --level=1 --device=md0 raid.01 raid.02 raid.03
raid /usr --level=5 --device=md1 raid.11 raid.12 raid.13
```

For a detailed example of **raid** in action, refer to Section 31.4.1, “Advanced Partitioning Example”.

reboot (optional)

Reboot after the installation is successfully completed (no arguments). Normally, kickstart displays a message and waits for the user to press a key before rebooting.

The **reboot** option is roughly equivalent to the **shutdown -r** command.

Specify **reboot** to automate installation fully when installing in cmdline mode on System z.

For other completion methods, refer to the **halt**, **poweroff**, and **shutdown** kickstart options.

The **halt** option is the default completion method if no other methods are explicitly specified in the kickstart file.

Note

Use of the **reboot** option *may* result in an endless installation loop, depending on the installation media and method.

repo (optional)

Configures additional yum repositories that may be used as sources for package installation. Multiple repo lines may be specified.

```
repo --name=<repoid> [--baseurl=<url>| --mirrorlist=<url>]
```

- **--name=** — The repo id. This option is required.
- **--baseurl=** — The URL for the repository. The variables that may be used in yum repo config files are not supported here. You may use one of either this option or --mirrorlist, not both.

- **--mirrorlist=** — The URL pointing at a list of mirrors for the repository. The variables that may be used in yum repo config files are not supported here. You may use one of either this option or --baseurl, not both.

rootpw (required)

Sets the system's root password to the *<password>* argument.

```
rootpw [--iscrypted] <password>
```

- **--iscrypted** — If this is present, the password argument is assumed to already be encrypted.

selinux (optional)

Sets the state of SELinux on the installed system. SELinux defaults to enforcing in anaconda.

```
selinux [--disabled|--enforcing|--permissive]
```

- **--enforcing** — Enables SELinux with the default targeted policy being enforced.

> **Note**
>
> If the **selinux** option is not present in the kickstart file, SELinux is enabled and set to **--enforcing** by default.

- **--permissive** — Outputs warnings based on the SELinux policy, but does not actually enforce the policy.

- **--disabled** — Disables SELinux completely on the system.

services (optional)

Modifies the default set of services that will run under the default runlevel. The services listed in the disabled list will be disabled before the services listed in the enabled list are enabled.

- **--disabled** — Disable the services given in the comma separated list.

- **--enabled** — Enable the services given in the comma separated list.

> **Do not include spaces in the list of services**
>
> If you include spaces in the comma-separated list, kickstart will enable or disable only the services up to the first space. For example:
>
> **`services --disabled auditd, cups,smartd, nfslock`**
>
> will disable only the **auditd** service. To disable all four services, this entry should include no spaces between services:
>
> **`services --disabled auditd,cups,smartd,nfslock`**

shutdown (optional)

Shut down the system after the installation has successfully completed. During a kickstart installation, if no completion method is specified, the **halt** option is used as default.

The **shutdown** option is roughly equivalent to the **shutdown** command.

For other completion methods, refer to the **halt**, **poweroff**, and **reboot** kickstart options.

skipx (optional)

If present, X is not configured on the installed system.

text (optional)

Perform the kickstart installation in text mode. Kickstart installations are performed in graphical mode by default.

timezone (required)

Sets the system time zone to *<timezone>* which may be any of the time zones listed by **timeconfig**.

```
timezone [--utc] <timezone>
```

- **--utc** — If present, the system assumes the hardware clock is set to UTC (Greenwich Mean) time.

upgrade (optional)

Tells the system to upgrade an existing system rather than install a fresh system. You must specify one of **cdrom**, **harddrive**, **nfs**, or **url** (for FTP and HTTP) as the location of the installation tree. Refer to **install** for details.

user (optional)

Creates a new user on the system.

```
user --name=<username> [--groups=<list>] [--homedir=<homedir>] [--
password=<password>] [--iscrypted] [--shell=<shell>] [--uid=<uid>]
```

- **--name=** — Provides the name of the user. This option is required.

- **--groups=** — In addition to the default group, a comma separated list of group names the user should belong to. The groups must exist before the user account is created.
- **--homedir=** — The home directory for the user. If not provided, this defaults to /home/<*username*>.
- **--password=** — The new user's password. If not provided, the account will be locked by default.
- **--iscrypted=** — Is the password provided by --password already encrypted or not?
- **--shell=** — The user's login shell. If not provided, this defaults to the system default.
- **--uid=** — The user's UID. If not provided, this defaults to the next available non-system UID.

vnc (optional)

Allows the graphical installation to be viewed remotely via VNC. This method is usually preferred over text mode, as there are some size and language limitations in text installs. With no options, this command will start a VNC server on the machine with no password and will print out the command that needs to be run to connect a remote machine.

```
vnc [--host=<hostname>] [--port=<port>] [--password=<password>]
```

- **--host=** — Instead of starting a VNC server on the install machine, connect to the VNC viewer process listening on the given hostname.
- **--port=** — Provide a port that the remote VNC viewer process is listening on. If not provided, anaconda will use the VNC default.
- **--password=** — Set a password which must be provided to connect to the VNC session. This is optional, but recommended.

volgroup (optional)

Use to create a Logical Volume Management (LVM) group with the syntax:

```
volgroup <name> <partition> <options>
```

The partition is described in the form **pv.<*id*>**. Any value can be assigned for <*id*> fields, as long as these values are consistent across volumes and volume groups. The default and lowest value is 01.

The options are as follows:

- **--noformat** — Use an existing volume group and do not format it.
- **--useexisting** — Use an existing volume group and reformat it.
- **--pesize=** — Set the size of the physical extents.

Create the partition first, create the logical volume group, and then create the logical volume. For example:

```
part pv.01 --size 3000
volgroup myvg pv.01
logvol / --vgname=myvg --size=2000 --name=rootvol
```

For a detailed example of **volgroup** in action, refer to Section 31.4.1, “Advanced Partitioning Example”.

xconfig (optional)

Configures the X Window System. If this option is not given, the user must configure X manually during the installation, if X was installed; this option should not be used if X is not installed on the final system.

- **--driver** — Specify the X driver to use for the video hardware.
- **--videoram=** — Specifies the amount of video RAM the video card has.
- **--defaultdesktop=** — Specify either GNOME or KDE to set the default desktop (assumes that GNOME Desktop Environment and/or KDE Desktop Environment has been installed through **%packages**).
- **--startxonboot** — Use a graphical login on the installed system.
- **--resolution=** — Specify the default resolution for the X Window System on the installed system. Valid values are 640x480, 800x600, 1024x768, 1152x864, 1280x1024, 1400x1050, 1600x1200. Be sure to specify a resolution that is compatible with the video card and monitor.
- **--depth=** — Specify the default color depth for the X Window System on the installed system. Valid values are 8, 16, 24, and 32. Be sure to specify a color depth that is compatible with the video card and monitor.

zerombr (optional)

If **zerombr** is specified any invalid partition tables found on disks are initialized. This destroys all of the contents of disks with invalid partition tables.

Note that this command was previously specified as **zerombr yes**. This form is now deprecated; you should now simply specify **zerombr** in your kickstart file instead.

zfcp (optional)

Define a Fiber channel device (IBM System z).

zfcp [--devnum=<*devnum*>] [--fcplun=<*fcplun*>] [--scsiid=<*scsiid*>] [--scsilun=<*scsilun*>] [--wwpn=<*wwpn*>]

%include (optional)

Use the **%include */path/to/file*** command to include the contents of another file in the kickstart file as though the contents were at the location of the **%include** command in the kickstart file.

31.4.1. Advanced Partitioning Example

The following is a single, integrated example showing the **clearpart**, **raid**, **part**, **volgroup**, and **logvol** kickstart options in action:

```
clearpart --drives=hda,hdc --initlabel
# Raid 1 IDE config
part raid.11    --size 1000     --asprimary     --ondrive=hda
part raid.12    --size 1000     --asprimary     --ondrive=hda
```

```
part raid.13    --size 2000     --asprimary     --ondrive=hda
part raid.14    --size 8000                     --ondrive=hda
part raid.15    --size 1 --grow                 --ondrive=hda
part raid.21    --size 1000     --asprimary     --ondrive=hdc
part raid.22    --size 1000     --asprimary     --ondrive=hdc
part raid.23    --size 2000     --asprimary     --ondrive=hdc
part raid.24    --size 8000                     --ondrive=hdc
part raid.25    --size 1 --grow                 --ondrive=hdc

# You can add --spares=x
raid /          --fstype ext3 --device md0 --level=RAID1 raid.11 raid.21
raid /safe      --fstype ext3 --device md1 --level=RAID1 raid.12 raid.22
raid swap       --fstype swap --device md2 --level=RAID1 raid.13 raid.23
raid /usr       --fstype ext3 --device md3 --level=RAID1 raid.14 raid.24
raid pv.01      --fstype ext3 --device md4 --level=RAID1 raid.15 raid.25

# LVM configuration so that we can resize /var and /usr/local later
volgroup sysvg pv.01
logvol /var             --vgname=sysvg  --size=8000     --name=var
logvol /var/freespace   --vgname=sysvg  --size=8000     --name=freespacetouse
logvol /usr/local       --vgname=sysvg  --size=1 --grow --name=usrlocal
```

This advanced example implements LVM over RAID, as well as the ability to resize various directories for future growth.

31.5. Package Selection

Warning — @Everything is not supported

You can use a kickstart file to install every available package by specifying **`@Everything`** or simply `*` in the **`%packages`** section. Red Hat does not support this type of installation.

Moreover, using a kickstart file in this way will introduce package and file conflicts onto the installed system. Packages known to cause such problems are assigned to the **`@Conflicts`** group. If you specify **`@Everything`** in a kickstart file, be sure to exclude **`@Conflicts`** or the installation will fail:

```
@Everything
-@Conflicts
```

Note that Red Hat does not support the use of **`@Everything`** in a kickstart file, even if you exclude **`@Conflicts`**.

Use the **`%packages`** command to begin a kickstart file section that lists the packages you would like to install (this is for installations only, as package selection during upgrades is not supported).

Packages can be specified by group or by individual package name, including with globs using the asterisk. The installation program defines several groups that contain related packages. Refer to the ***`variant`*****`/repodata/comps-*.xml`** file on the first Red Hat Enterprise Linux CD-ROM for a list of groups. Each group has an id, user visibility value, name, description, and package list. In the package list, the packages marked as mandatory are always installed if the group is selected, the packages marked default are selected by default if the group is selected, and the packages marked optional must be specifically selected even if the group is selected to be installed.

Available groups vary slightly between different variants of Red Hat Enterprise Linux 5, but include:

- Administration Tools
- Authoring and Publishing
- Development Libraries
- Development Tools
- DNS Name Server
- Eclipse
- Editors
- Engineering and Scientific
- FTP Server
- GNOME Desktop Environment
- GNOME Software Development
- Games and Entertainment
- Graphical Internet
- Graphics
- Java Development
- KDE (K Desktop Environment)
- KDE Software Development
- Legacy Network Server
- Legacy Software Development
- Legacy Software Support
- Mail Server
- Misc
- Multimedia
- MySQL Database
- Network Servers
- News Server
- Office/Productivity
- OpenFabrics Enterprise Distribution
- PostgreSQL Database
- Printing Support
- Server Configuration Tools
- Sound and Video

- System Tools
- Text-based Internet
- Web Server
- Windows File Server
- Windows PV Drivers
- X Software Development
- X Window System

In most cases, it is only necessary to list the desired groups and not individual packages. Note that the **Core** and **Base** groups are always selected by default, so it is not necessary to specify them in the **%packages** section.

Here is an example **%packages** selection:

```
%packages
@ X Window System
@ GNOME Desktop Environment
@ Graphical Internet
@ Sound and Video dhcp
```

As you can see, groups are specified, one to a line, starting with an **@** symbol, a space, and then the full group name as given in the **comps.xml** file. Groups can also be specified using the id for the group, such as **gnome-desktop**. Specify individual packages with no additional characters (the **dhcp** line in the example above is an individual package).

You can also specify which packages not to install from the default package list:

```
-autofs
```

The following options are available for the **%packages** option:

--nobase

Do not install the @Base group. Use this option if you are trying to create a very small system.

--resolvedeps

The --resolvedeps option has been deprecated. Dependencies are resolved automatically every time now.

--ignoredeps

The --ignoredeps option has been deprecated. Dependencies are resolved automatically every time now.

--ignoremissing

Ignore the missing packages and groups instead of halting the installation to ask if the installation should be aborted or continued. For example:

```
%packages --ignoremissing
```

31.6. Pre-installation Script

You can add commands to run on the system immediately after the **ks.cfg** has been parsed. This section must be at the end of the kickstart file (after the commands) and must start with the **%pre** command. You can access the network in the **%pre** section; however, *name service* has not been configured at this point, so only IP addresses work.

> **Note**
>
> Note that the pre-install script is not run in the change root environment.

--interpreter ***/usr/bin/python***

Allows you to specify a different scripting language, such as Python. Replace */usr/bin/python* with the scripting language of your choice.

31.6.1. Example

Here is an example **%pre** section:

```
%pre
#!/bin/sh
hds=""
mymedia=""
for file in /proc/ide/h* do
 mymedia=`cat $file/media`
 if [ $mymedia == "disk" ] ; then
  hds="$hds `basename $file`"
 fi
done
set $hds
numhd=`echo $#`
drive1=`echo $hds | cut -d' ' -f1`
drive2=`echo $hds | cut -d' ' -f2`
#Write out partition scheme based on whether there are 1 or 2 hard drives
if [ $numhd == "2" ] ; then
 #2 drives
 echo "#partitioning scheme generated in %pre for 2 drives" > /tmp/part-include
 echo "clearpart --all" >> /tmp/part-include
 echo "part /boot --fstype ext3 --size 75 --ondisk hda" >> /tmp/part-include
 echo "part / --fstype ext3 --size 1 --grow --ondisk hda" >> /tmp/part-include
 echo "part swap --recommended --ondisk $drive1" >> /tmp/part-include
 echo "part /home --fstype ext3 --size 1 --grow --ondisk hdb" >> /tmp/part-include
else
 #1 drive
 echo "#partitioning scheme generated in %pre for 1 drive" > /tmp/part-include
 echo "clearpart --all" >> /tmp/part-include
 echo "part /boot --fstype ext3 --size 75" >> /tmp/part-include
 echo "part swap --recommended" >> /tmp/part-include
 echo "part / --fstype ext3 --size 2048" >> /tmp/part-include
 echo "part /home --fstype ext3 --size 2048 --grow" >> /tmp/part-include
fi
```

This script determines the number of hard drives in the system and writes a text file with a different partitioning scheme depending on whether it has one or two drives. Instead of having a set of partitioning commands in the kickstart file, include the line:

```
%include /tmp/part-include
```

The partitioning commands selected in the script are used.

> **Note**
>
> The pre-installation script section of kickstart *cannot* manage multiple install trees or source media. This information must be included for each created ks.cfg file, as the pre-installation script occurs during the second stage of the installation process.

31.7. Post-installation Script

You have the option of adding commands to run on the system once the installation is complete. This section must be at the end of the kickstart file and must start with the **%post** command. This section is useful for functions such as installing additional software and configuring an additional nameserver.

> **Note**
>
> If you configured the network with static IP information, including a nameserver, you can access the network and resolve IP addresses in the **%post** section. If you configured the network for DHCP, the **/etc/resolv.conf** file has not been completed when the installation executes the **%post** section. You can access the network, but you can not resolve IP addresses. Thus, if you are using DHCP, you must specify IP addresses in the **%post** section.

> **Note**
>
> The post-install script is run in a chroot environment; therefore, performing tasks such as copying scripts or RPMs from the installation media do not work.

--nochroot

Allows you to specify commands that you would like to run outside of the chroot environment.

The following example copies the file **/etc/resolv.conf** to the file system that was just installed.

```
%post --nochroot
cp /etc/resolv.conf /mnt/sysimage/etc/resolv.conf
```

--interpreter *\/usr/bin/python*

Allows you to specify a different scripting language, such as Python. Replace */usr/bin/python* with the scripting language of your choice.

--log */path/to/logfile*

Logs the output of the post-install script. Note that the path of the log file must take into account whether or not you use the **--nochroot** option. For example, without **--**

nochroot:

This command is available in Red Hat Enterprise Linux 5.5 and later.

```
%post --log=/root/ks-post.log
```

with **--nochroot**:

```
%post --nochroot --log=/mnt/sysimage/root/ks-post.log
```

31.7.1. Examples

Register the system to a Subscription Asset Manager server, using the **--log** option to log the result (in Red Hat Enterprise Linux 5.5 and later):

```
%post --log=/root/ks-post.log
/usr/sbin/subscription-manager register --username=admin@example.com --
password=secret --serverurl=sam-server.example.com --org="Admin Group" --
environment="Dev" --servicelevel=standard
```

Run a script named **runme** from an NFS share:

```
mkdir /mnt/temp
mount -o nolock 10.10.0.2:/usr/new-machines /mnt/temp open -s -w --
/mnt/temp/runme
umount /mnt/temp
```

Note

NFS file locking is *not* supported while in kickstart mode, therefore **-o nolock** is required when mounting an NFS mount.

31.8. Making the Kickstart File Available

A kickstart file must be placed in one of the following locations:

- On a boot diskette
- On a boot CD-ROM
- On a network

Normally a kickstart file is copied to the boot diskette, or made available on the network. The network-based approach is most commonly used, as most kickstart installations tend to be performed on networked computers.

Let us take a more in-depth look at where the kickstart file may be placed.

31.8.1. Creating Kickstart Boot Media

Diskette-based booting is no longer supported in Red Hat Enterprise Linux. Installations must use CD-ROM or flash memory products for booting. However, the kickstart file may still reside on a diskette's top-level directory, and must be named **ks.cfg**.

To perform a CD-ROM-based kickstart installation, the kickstart file must be named **ks.cfg** and must be located in the boot CD-ROM's top-level directory. Since a CD-ROM is read-only, the file must be added to the directory used to create the image that is written to the CD-ROM. Refer to Section 2.4.1, “Alternative Boot Methods” for instructions on creating boot media; however, before making the **file.iso** image file, copy the **ks.cfg** kickstart file to the **isolinux/** directory.

To perform a pen-based flash memory kickstart installation, the kickstart file must be named **ks.cfg** and must be located in the flash memory's top-level directory. Create the boot image first, and then copy the **ks.cfg** file.

> **Note**
>
> Creation of USB flash memory pen drives for booting is possible, but is heavily dependent on system hardware BIOS settings. Refer to your hardware manufacturer to see if your system supports booting to alternate devices.

31.8.2. Making the Kickstart File Available on the Network

Network installations using kickstart are quite common, because system administrators can easily automate the installation on many networked computers quickly and painlessly. In general, the approach most commonly used is for the administrator to have both a BOOTP/DHCP server and an NFS server on the local network. The BOOTP/DHCP server is used to give the client system its networking information, while the actual files used during the installation are served by the NFS server. Often, these two servers run on the same physical machine, but they are not required to.

To perform a network-based kickstart installation, you must have a BOOTP/DHCP server on your network, and it must include configuration information for the machine on which you are attempting to install Red Hat Enterprise Linux. The BOOTP/DHCP server provides the client with its networking information as well as the location of the kickstart file.

If a kickstart file is specified by the BOOTP/DHCP server, the client system attempts an NFS mount of the file's path, and copies the specified file to the client, using it as the kickstart file. The exact settings required vary depending on the BOOTP/DHCP server you use.

Here is an example of a line from the **dhcpd.conf** file for the DHCP server:

```
filename "/usr/new-machine/kickstart/"; next-server blarg.redhat.com;
```

Note that you should replace the value after **filename** with the name of the kickstart file (or the directory in which the kickstart file resides) and the value after **next-server** with the NFS server name.

If the file name returned by the BOOTP/DHCP server ends with a slash ("/"), then it is interpreted as a path only. In this case, the client system mounts that path using NFS, and searches for a particular file. The file name the client searches for is:

```
<ip-addr>-kickstart
```

The ***<ip-addr>*** section of the file name should be replaced with the client's IP address in dotted decimal notation. For example, the file name for a computer with an IP address of 10.10.0.1 would be **10.10.0.1-kickstart**.

Note that if you do not specify a server name, then the client system attempts to use the server that answered the BOOTP/DHCP request as its NFS server. If you do not specify a path or file name, the client system tries to mount **`/kickstart`** from the BOOTP/DHCP server and tries to find the kickstart file using the same **`<ip-addr>-kickstart`** file name as described above.

31.9. Making the Installation Tree Available

The kickstart installation must access an *installation tree*. An installation tree is a copy of the binary Red Hat Enterprise Linux CD-ROMs with the same directory structure.

If you are performing a CD-based installation, insert the Red Hat Enterprise Linux CD-ROM #1 into the computer before starting the kickstart installation.

If you are performing a hard drive installation, make sure the ISO images of the binary Red Hat Enterprise Linux CD-ROMs are on a hard drive in the computer.

If you are performing a network-based (NFS, FTP, or HTTP) installation, you must make the installation tree available over the network. Refer to Section 2.5, “Preparing for a Network Installation” for details.

31.10. Starting a Kickstart Installation

To begin a kickstart installation, you must boot the system from boot media you have made or the Red Hat Enterprise Linux CD-ROM #1, and enter a special boot command at the boot prompt. The installation program looks for a kickstart file if the **`ks`** command line argument is passed to the kernel.

CD-ROM #1 and Diskette

The **`linux ks=floppy`** command also works if the **`ks.cfg`** file is located on a vfat or ext2 file system on a diskette and you boot from the Red Hat Enterprise Linux CD-ROM #1.

An alternate boot command is to boot off the Red Hat Enterprise Linux CD-ROM #1 and have the kickstart file on a vfat or ext2 file system on a diskette. To do so, enter the following command at the **`boot:`** prompt:

```
linux ks=hd:fd0:/ks.cfg
```

With Driver Disk

If you need to use a driver disk with kickstart, specify the **`dd`** option as well. For example, to boot off a boot diskette and use a driver disk, enter the following command at the **`boot:`** prompt:

```
linux ks=floppy dd
```

Boot CD-ROM

If the kickstart file is on a boot CD-ROM as described in Section 31.8.1, “Creating Kickstart Boot Media”, insert the CD-ROM into the system, boot the system, and enter the following command at the **`boot:`** prompt (where **`ks.cfg`** is the name of the kickstart file):

```
linux ks=cdrom:/ks.cfg
```

Other options to start a kickstart installation are as follows:

askmethod

Do not automatically use the CD-ROM as the install source if we detect a Red Hat Enterprise Linux CD in your CD-ROM drive.

autostep

Make kickstart non-interactive. Used for debugging and to generate screenshots. This option should not be used when deploying a system because it may disrupt package installation.

debug

Start up pdb immediately.

dd

Use a driver disk.

dhcpclass=<*class*>

Sends a custom DHCP vendor class identifier. ISC's dhcpcd can inspect this value using "option vendor-class-identifier".

dns=<*dns*>

Comma separated list of nameservers to use for a network installation.

driverdisk

Same as 'dd'.

expert

Turns on special features:

- allows partitioning of removable media
- prompts for a driver disk

gateway=<*gw*>

Gateway to use for a network installation.

graphical

Force graphical install. Required to have ftp/http use GUI.

isa

Prompt user for ISA devices configuration.

ip=<*ip*>

IP to use for a network installation, use 'dhcp' for DHCP.

keymap=<*keymap*>

Keyboard layout to use. Valid values are those which can be used for the 'keyboard' kickstart command.

ks=nfs:<*server*>:/<*path*>

The installation program looks for the kickstart file on the NFS server <*server*>, as file <*path*>. The installation program uses DHCP to configure the Ethernet card. For example, if your NFS server is server.example.com and the kickstart file is in the NFS share **/mydir/ks.cfg**, the correct boot command would be **ks=nfs:server.example.com:/mydir/ks.cfg**.

ks=http://<*server*>/<*path*>

The installation program looks for the kickstart file on the HTTP server <*server*>, as file <*path*>. The installation program uses DHCP to configure the Ethernet card. For example, if your HTTP server is server.example.com and the kickstart file is in the HTTP directory **/mydir/ks.cfg**, the correct boot command would be **ks=http://server.example.com/mydir/ks.cfg**.

ks=floppy

The installation program looks for the file **ks.cfg** on a vfat or ext2 file system on the diskette in **/dev/fd0**.

ks=floppy:/<*path*>

The installation program looks for the kickstart file on the diskette in **/dev/fd0**, as file <*path*>.

ks=hd:<*device*>:/<*file*>

The installation program mounts the file system on <*device*> (which must be vfat or ext2), and look for the kickstart configuration file as <*file*> in that file system (for example, **ks=hd:sda3:/mydir/ks.cfg**).

ks=file:/<*file*>

The installation program tries to read the file <*file*> from the file system; no mounts are done. This is normally used if the kickstart file is already on the **initrd** image.

ks=cdrom:/<*path*>

The installation program looks for the kickstart file on CD-ROM, as file <*path*>.

ks

If **ks** is used alone, the installation program configures the Ethernet card to use DHCP. The kickstart file is read from the "bootServer" from the DHCP response as if it is an NFS server sharing the kickstart file. By default, the bootServer is the same as the DHCP server. The name of the kickstart file is one of the following:

- If DHCP is specified and the boot file begins with a **/**, the boot file provided by DHCP is looked for on the NFS server.
- If DHCP is specified and the boot file begins with something other than a **/**, the boot file provided by DHCP is looked for in the **/kickstart** directory on the NFS server.
- If DHCP did not specify a boot file, then the installation program tries to read the file **/kickstart/1.2.3.4-kickstart**, where *1.2.3.4* is the numeric IP address of the machine being installed.

ksdevice=<*device*>

The installation program uses this network device to connect to the network. For example, consider a system connected to an NFS server through the eth1 device. To perform a kickstart installation on this system using a kickstart file from the NFS server, you would use the command **ks=nfs:<*server*>:/<*path*> ksdevice=eth1** at the **boot:** prompt.

kssendmac

Adds HTTP headers to ks=http:// request that can be helpful for provisioning systems. Includes MAC address of all nics in CGI environment variables of the form: "X-RHN-Provisioning-MAC-0: eth0 01:23:45:67:89:ab".

lang=<*lang*>

Language to use for the installation. This should be a language which is valid to be used with the 'lang' kickstart command.

loglevel=<*level*>

Set the minimum level required for messages to be logged. Values for <level> are debug, info, warning, error, and critical. The default value is info.

lowres

Force GUI installer to run at 640x480.

mediacheck

Activates loader code to give user option of testing integrity of install source (if an ISO-based method).

method=cdrom://

Do a CDROM based installation.

method=ftp://<*path*>

Use <path> for an FTP installation.

method=hd:<*dev*>:<*path*>

Use <path> on <dev> for a hard drive installation.

method=http://<*path*>

Use <path> for an HTTP installation.

method=nfs:<*path*>

Use <path> for an NFS installation.

netmask=<*nm*>

Netmask to use for a network installation.

nofallback

If GUI fails exit.

nofb

Do not load the VGA16 framebuffer required for doing text-mode installation in some languages.

`nofirewire`

Do not load support for firewire devices.

`noipv6`

Disable IPv6 networking during installation.

> **This option is not available during PXE installations**
>
> During installations from a PXE server, IPv6 networking might become active before **anaconda** processes the Kickstart file. If so, this option will have no effect during installation.

`nomount`

Don't automatically mount any installed Linux partitions in rescue mode.

`nonet`

Do not auto-probe network devices.

`noparport`

Do not attempt to load support for parallel ports.

`nopass`

Don't pass keyboard/mouse info to stage 2 installer, good for testing keyboard and mouse config screens in stage2 installer during network installs.

`nopcmcia`

Ignore PCMCIA controller in system.

`noprobe`

Do not attempt to detect hw, prompts user instead.

`noshell`

Do not put a shell on tty2 during install.

`nostorage`

Do not auto-probe storage devices (SCSI, IDE, RAID).

`nousb`

Do not load USB support (helps if install hangs early sometimes).

`nousbstorage`

Do not load usbstorage module in loader. May help with device ordering on SCSI systems.

`rescue`

Run rescue environment.

`resolution=<mode>`

Run installer in mode specified, '1024x768' for example.

`serial`

Turns on serial console support.

`skipddc`

Skips DDC probe of monitor, may help if it's hanging system.

`syslog=<host>[:<port>]`

Once installation is up and running, send log messages to the syslog process on *<host>*, and optionally, on port *<port>*. Requires the remote syslog process to accept connections (the -r option).

`text`

Force text mode install.

`updates`

Prompt for floppy containing updates (bug fixes).

`updates=ftp://<path>`

Image containing updates over FTP.

`updates=http://<path>`

Image containing updates over HTTP.

`upgradeany`

Don't require an /etc/redhat-release that matches the expected syntax to upgrade.

`vnc`

Enable vnc-based installation. You will need to connect to the machine using a vnc client application.

`vncconnect=<host>[:<port>]`

Once installation is up and running, connect to the vnc client named *<host>*, and optionally use port *<port>*.

Requires 'vnc' option to be specified as well.

`vncpassword=<password>`

Enable a password for the vnc connection. This will prevent someone from inadvertently connecting to the vnc-based installation.

Requires 'vnc' option to be specified as well.

Chapter 32. Kickstart Configurator

Kickstart Configurator allows you to create or modify a kickstart file using a graphical user interface, so that you do not have to remember the correct syntax of the file.

To use **Kickstart Configurator**, you must be running the X Window System and have **Kickstart Configurator** installed on your system. **Kickstart Configurator** is not installed by default, so you might need to install it with **yum** or your graphical package manager.

To start **Kickstart Configurator**, select **Applications** (the main menu on the panel) => **System Tools** => **Kickstart**, or type the command **`/usr/sbin/system-config-kickstart`**.

As you are creating a kickstart file, you can select **File** => **Preview** at any time to review your current selections.

To start with an existing kickstart file, select **File** => **Open** and select the existing file.

32.1. Basic Configuration

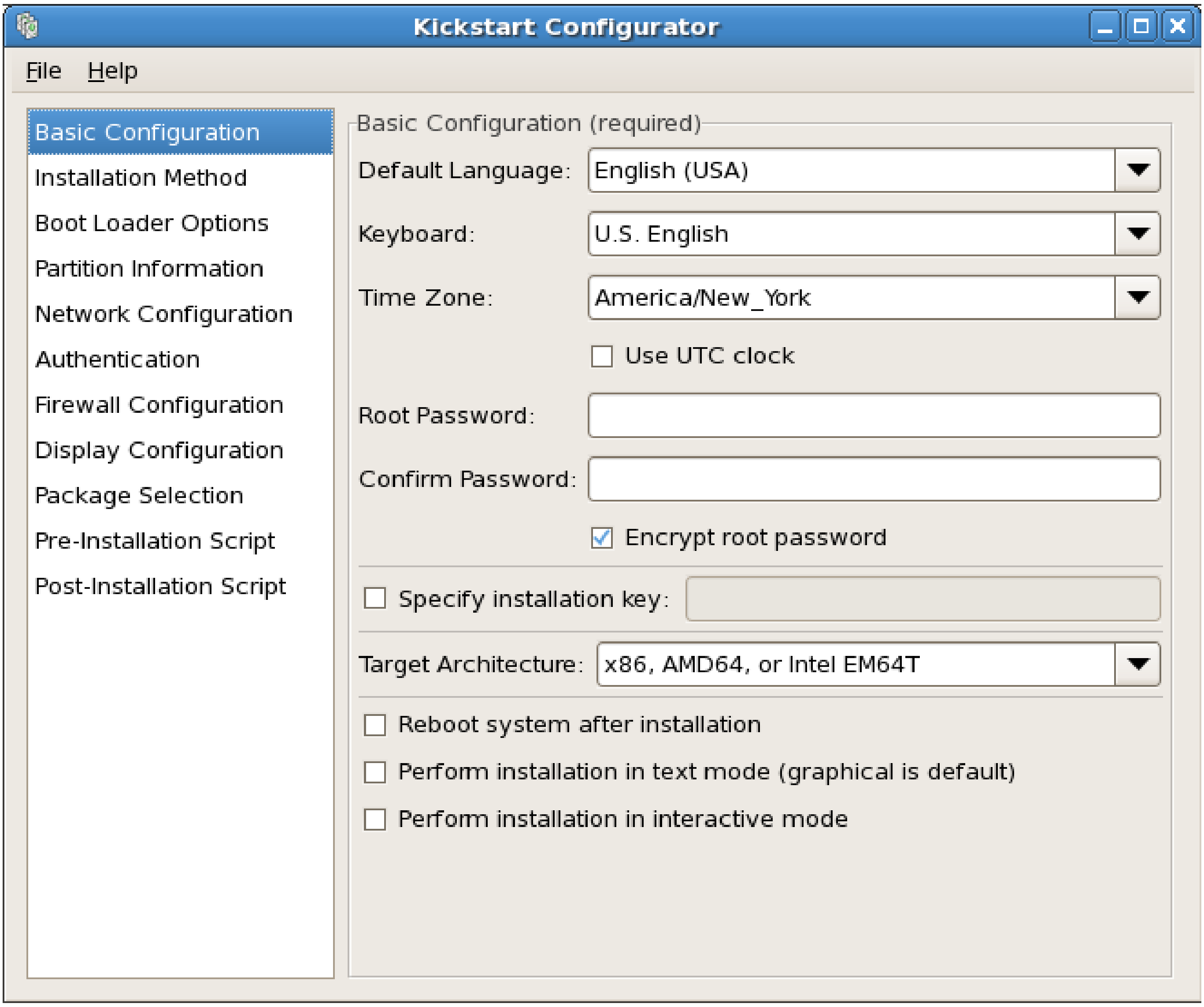

Figure 32.1. Basic Configuration

Choose the language to use during the installation and as the default language to be used after installation from the **Default Language** menu.

Select the system keyboard type from the **Keyboard** menu.

From the **Time Zone** menu, choose the time zone to use for the system. To configure the system to use UTC, select **Use UTC clock**.

Enter the desired root password for the system in the **Root Password** text entry box. Type the same password in the **Confirm Password** text box. The second field is to make sure you do not mistype the password and then realize you do not know what it is after you have completed the installation. To save the password as an encrypted password in the file, select **Encrypt root password**. If the encryption option is selected, when the file is saved, the plain text password that you typed is encrypted and written to the kickstart file. Do not type an already encrypted password and select to encrypt it. Because a kickstart file is a plain text file that can be easily read, it is recommended that an encrypted password be used.

Select the **Specify installation key** checkbox to provide an installation key.

Choosing **Target Architecture** specifies which specific hardware architecture distribution is used during installation.

Choosing **Reboot system after installation** reboots your system automatically after the installation is finished.

Kickstart installations are performed in graphical mode by default. To override this default and use text mode instead, select the **Perform installation in text mode** option.

You can perform a kickstart installation in interactive mode. This means that the installation program uses all the options pre-configured in the kickstart file, but it allows you to preview the options in each screen before continuing to the next screen. To continue to the next screen, click the **Next** button after you have approved the settings or change them before continuing the installation. To select this type of installation, select the **Perform installation in interactive mode** option.

32.2. Installation Method

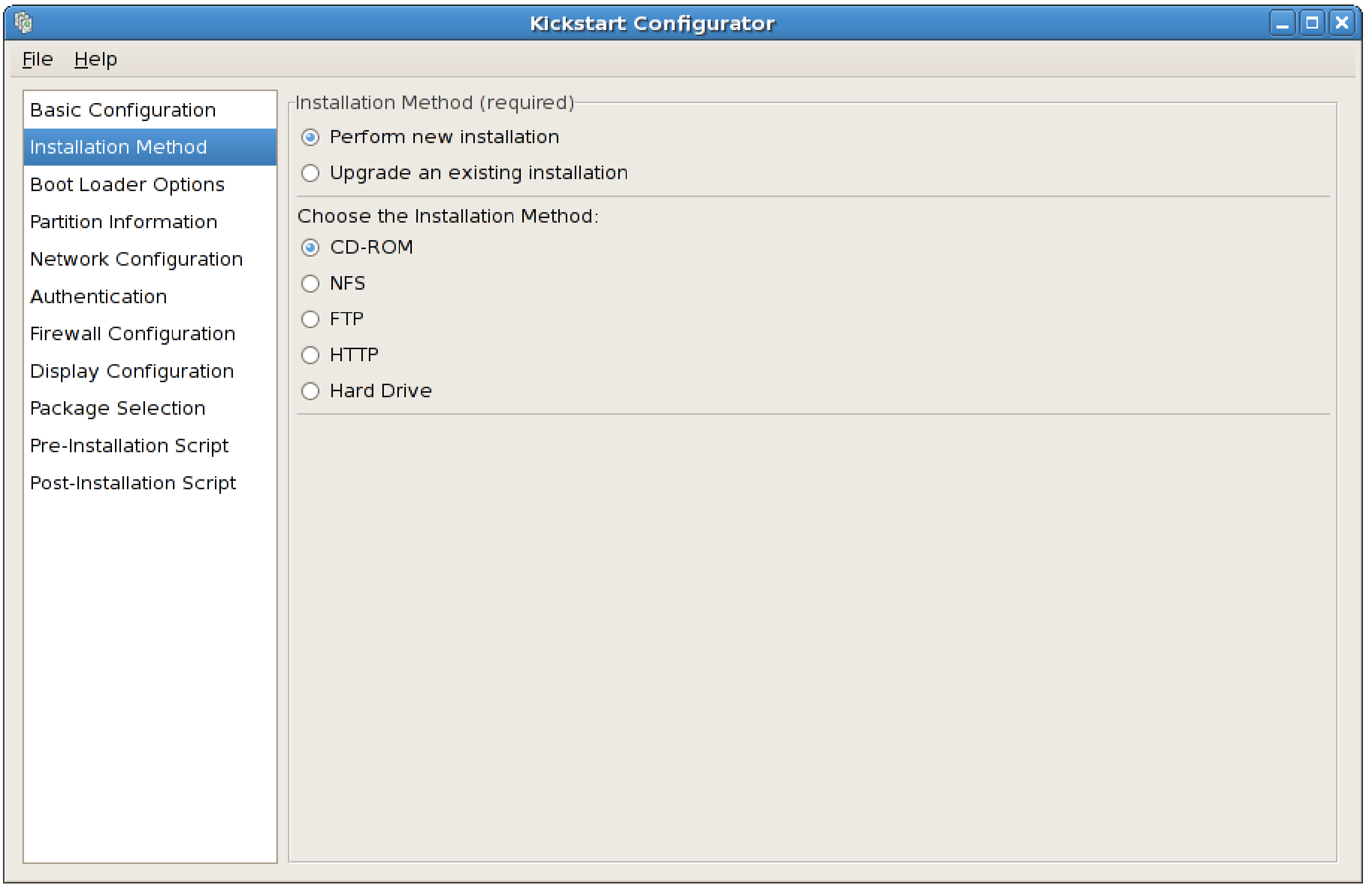

Figure 32.2. Installation Method

The **`Installation Method`** screen allows you to choose whether to perform a new installation or an upgrade. If you choose upgrade, the **`Partition Information`** and **`Package Selection`** options are disabled. They are not supported for kickstart upgrades.

Choose the type of kickstart installation or upgrade from the following options:

- **`CD-ROM`** — Choose this option to install or upgrade from the Red Hat Enterprise Linux CD-ROMs.
- **`NFS`** — Choose this option to install or upgrade from an NFS shared directory. In the text field for the NFS server, enter a fully-qualified domain name or IP address. For the NFS directory, enter the name of the NFS directory that contains the ***`variant`*** directory of the installation tree. For example, if the NFS server contains the directory **`/mirrors/redhat/i386/Server/`**, enter **`/mirrors/redhat/i386/`** for the NFS directory.
- **`FTP`** — Choose this option to install or upgrade from an FTP server. In the FTP server text field, enter a fully-qualified domain name or IP address. For the FTP directory, enter the name of the FTP directory that contains the ***`variant`*** directory. For example, if the FTP server contains the directory **`/mirrors/redhat/i386/Server/`**, enter **`/mirrors/redhat/i386/Server/`** for the FTP directory. If the FTP server requires a username and password, specify them as well.
- **`HTTP`** — Choose this option to install or upgrade from an HTTP server. In the text field for the HTTP server, enter the fully-qualified domain name or IP address. For the HTTP directory, enter the name of the HTTP directory that contains the ***`variant`*** directory. For example, if the HTTP server contains the directory **`/mirrors/redhat/i386/Server/`**, enter **`/mirrors/redhat/i386/Server/`** for the HTTP directory.
- **`Hard Drive`** — Choose this option to install or upgrade from a hard drive. Hard drive installations require the use of ISO (or CD-ROM) images. Be sure to verify that the ISO images are

intact before you start the installation. To verify them, use an **md5sum** program as well as the **linux mediacheck** boot option as discussed in the *Red Hat Enterprise Linux Installation Guide*. Enter the hard drive partition that contains the ISO images (for example, **/dev/hda1**) in the **Hard Drive Partition** text box. Enter the directory that contains the ISO images in the **Hard Drive Directory** text box.

32.3. Boot Loader Options

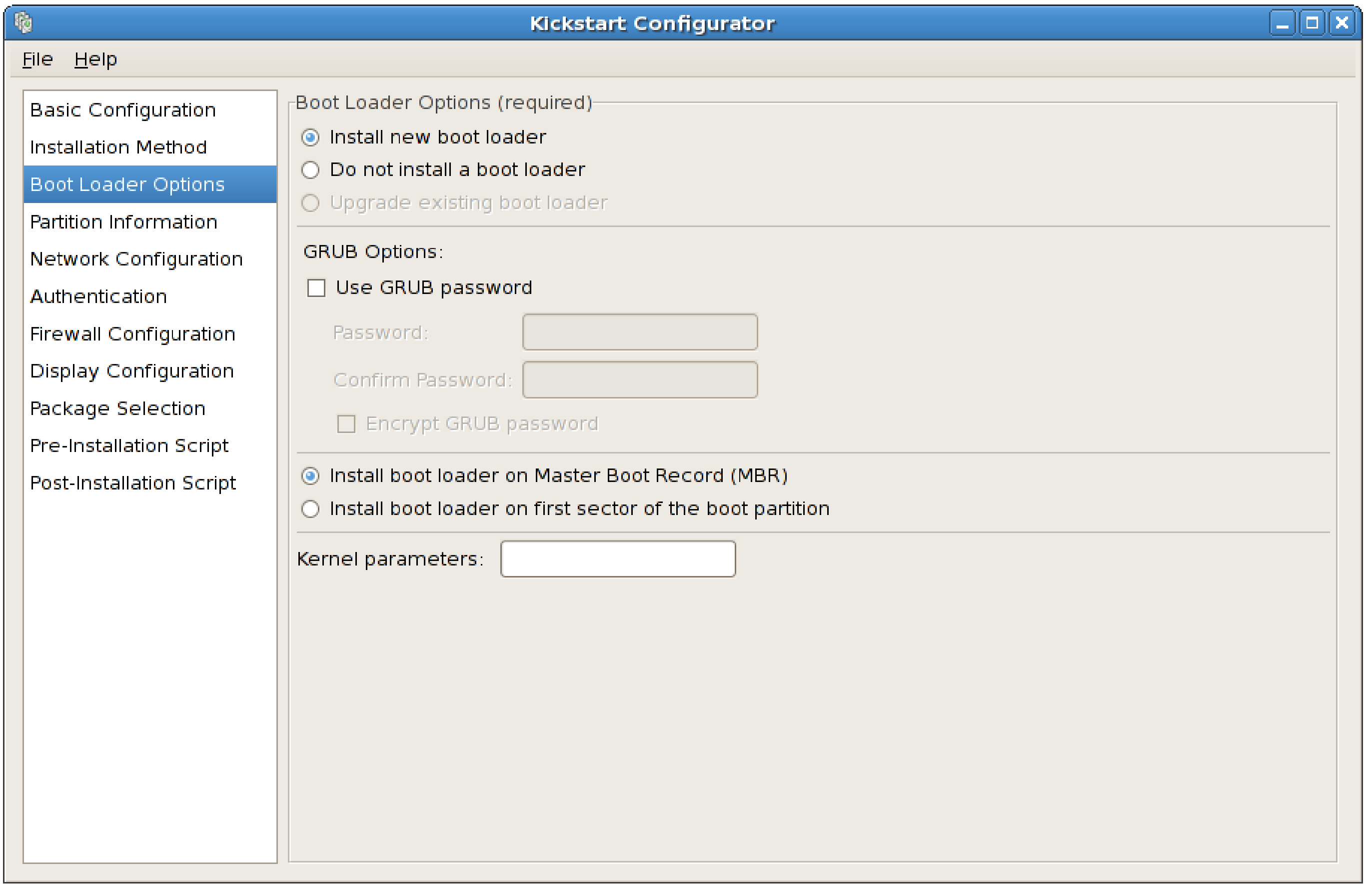

Figure 32.3. Boot Loader Options

Please note that this screen will be disabled if you have specified a target architecture other than x86 / x86_64.

GRUB is the default boot loader for Red Hat Enterprise Linux on x86 / x86_64 architectures. If you do not want to install a boot loader, select **Do not install a boot loader**. If you choose not to install a boot loader, make sure you create a boot diskette or have another way to boot your system, such as a third-party boot loader.

You must choose where to install the boot loader (the Master Boot Record or the first sector of the **/boot** partition). Install the boot loader on the MBR if you plan to use it as your boot loader.

To pass any special parameters to the kernel to be used when the system boots, enter them in the **Kernel parameters** text field. For example, if you have an IDE CD-ROM Writer, you can tell the kernel to use the SCSI emulation driver that must be loaded before using **cdrecord** by configuring **hdd=ide-scsi** as a kernel parameter (where **hdd** is the CD-ROM device).

You can password protect the GRUB boot loader by configuring a GRUB password. Select **Use GRUB password**, and enter a password in the **Password** field. Type the same password in the

Confirm Password text field. To save the password as an encrypted password in the file, select **Encrypt GRUB password**. If the encryption option is selected, when the file is saved, the plain text password that you typed is encrypted and written to the kickstart file. If the password you typed was already encrypted, unselect the encryption option.

If **Upgrade an existing installation** is selected on the **Installation Method** page, select **Upgrade existing boot loader** to upgrade the existing boot loader configuration, while preserving the old entries.

32.4. Partition Information

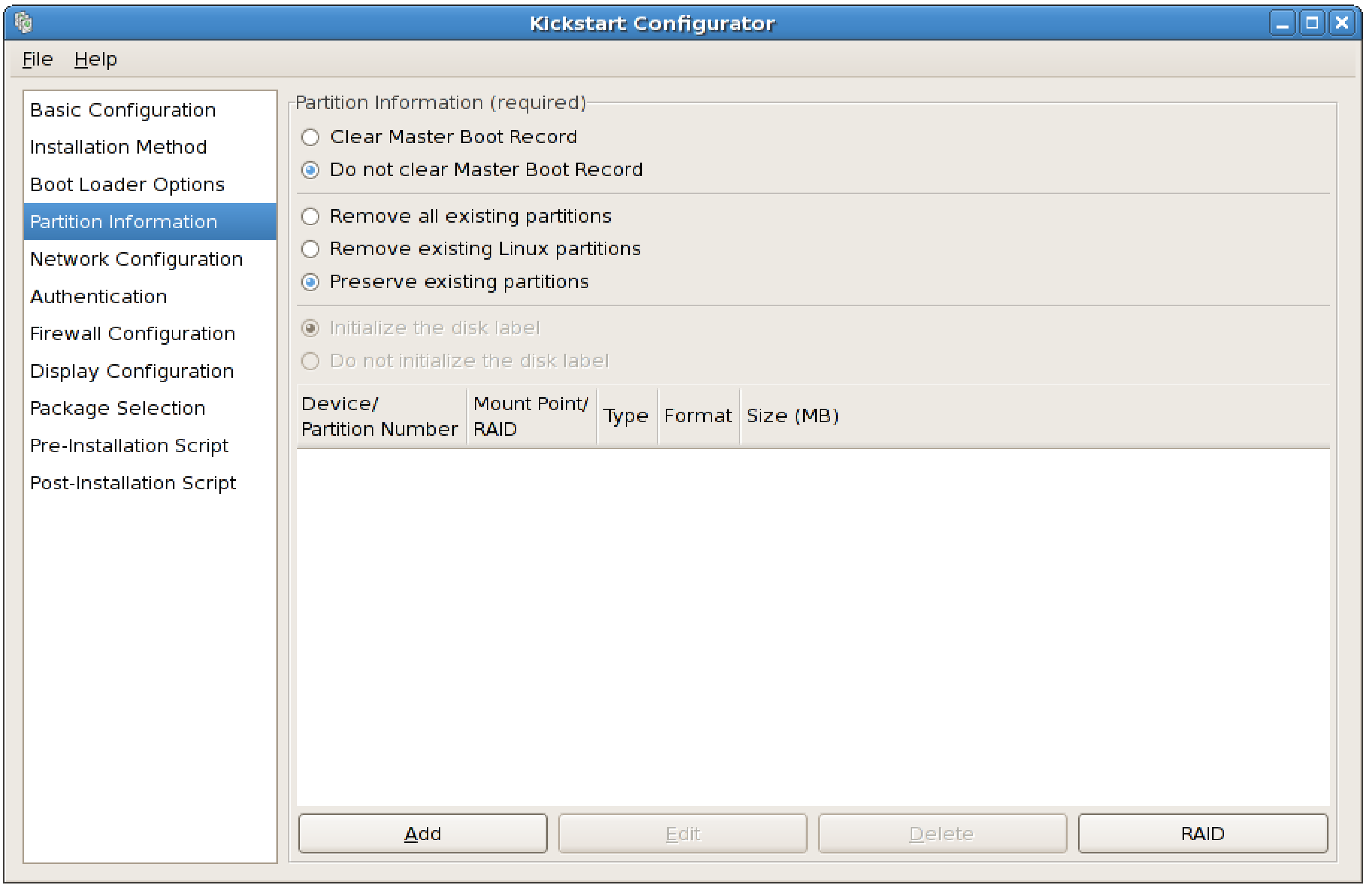

Figure 32.4. Partition Information

Select whether or not to clear the Master Boot Record (MBR). Choose to remove all existing partitions, remove all existing Linux partitions, or preserve existing partitions.

To initialize the disk label to the default for the architecture of the system (for example, **msdos** for x86 and **gpt** for Itanium), select **Initialize the disk label** if you are installing on a brand new hard drive.

Note

Although **anaconda** and **kickstart** support Logical Volume Management (LVM), at present there is no mechanism for configuring this using the **Kickstart Configurator**.

32.4.1. Creating Partitions

To create a partition, click the **Add** button. The **Partition Options** window shown in Figure 32.5, "Creating Partitions" appears. Choose the mount point, file system type, and partition size for the new partition. Optionally, you can also choose from the following:

- In the **Additional Size Options** section, choose to make the partition a fixed size, up to a chosen size, or fill the remaining space on the hard drive. If you selected swap as the file system type, you can select to have the installation program create the swap partition with the recommended size instead of specifying a size.
- Force the partition to be created as a primary partition.
- Create the partition on a specific hard drive. For example, to make the partition on the first IDE hard disk (**/dev/hda**), specify **hda** as the drive. Do not include **/dev** in the drive name.
- Use an existing partition. For example, to make the partition on the first partition on the first IDE hard disk (**/dev/hda1**), specify **hda1** as the partition. Do not include **/dev** in the partition name.
- Format the partition as the chosen file system type.

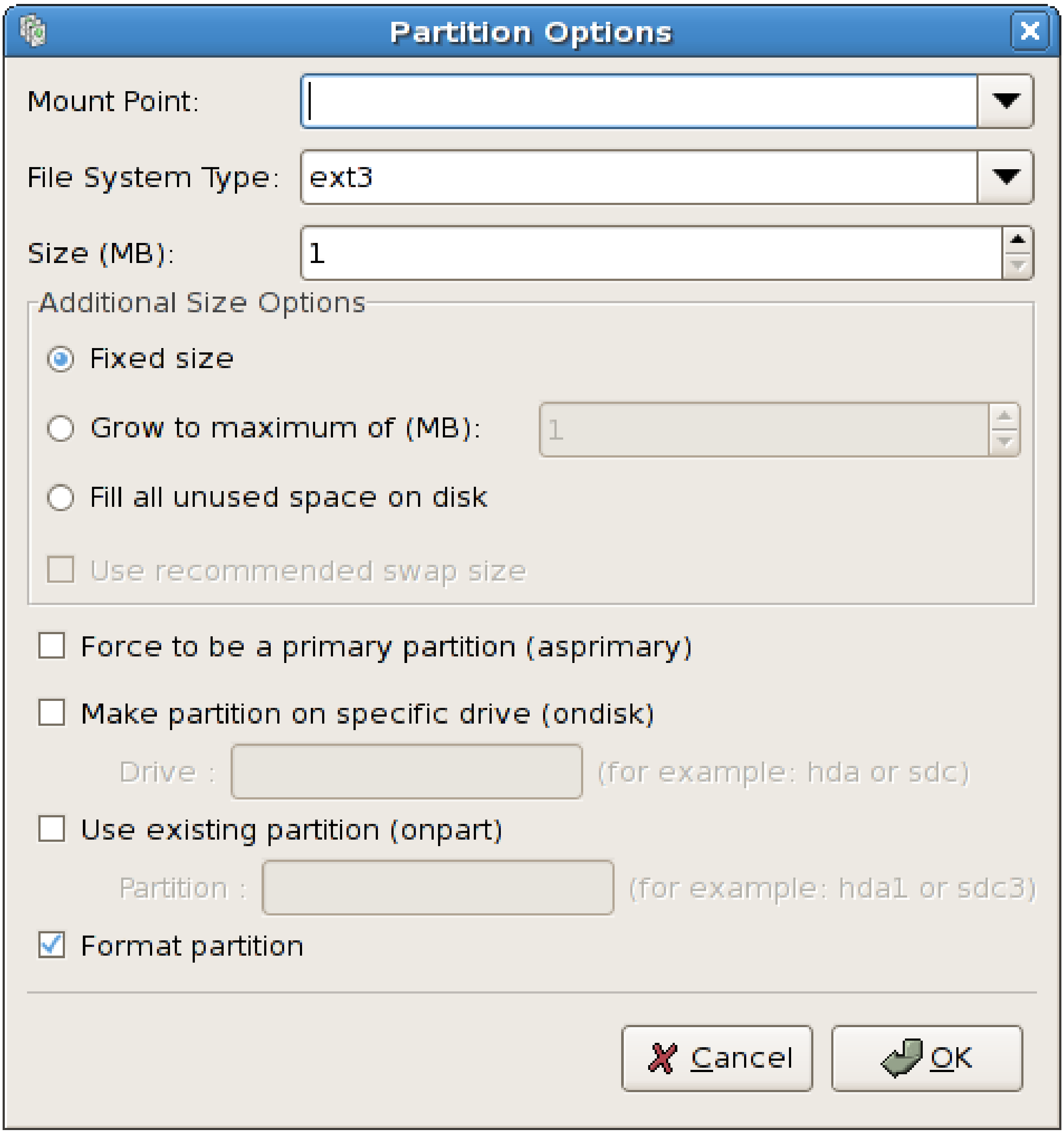

Figure 32.5. Creating Partitions

To edit an existing partition, select the partition from the list and click the **Edit** button. The same **Partition Options** window appears as when you chose to add a partition as shown in Figure 32.5, “Creating Partitions”, except it reflects the values for the selected partition. Modify the partition options and click **OK**.

To delete an existing partition, select the partition from the list and click the **Delete** button.

32.4.1.1. Creating Software RAID Partitions

To create a software RAID partition, use the following steps:

1. Click the **RAID** button.
2. Select **Create a software RAID partition**.

3. Configure the partitions as previously described, except select **`Software RAID`** as the file system type. Also, you must specify a hard drive on which to make the partition or specify an existing partition to use.

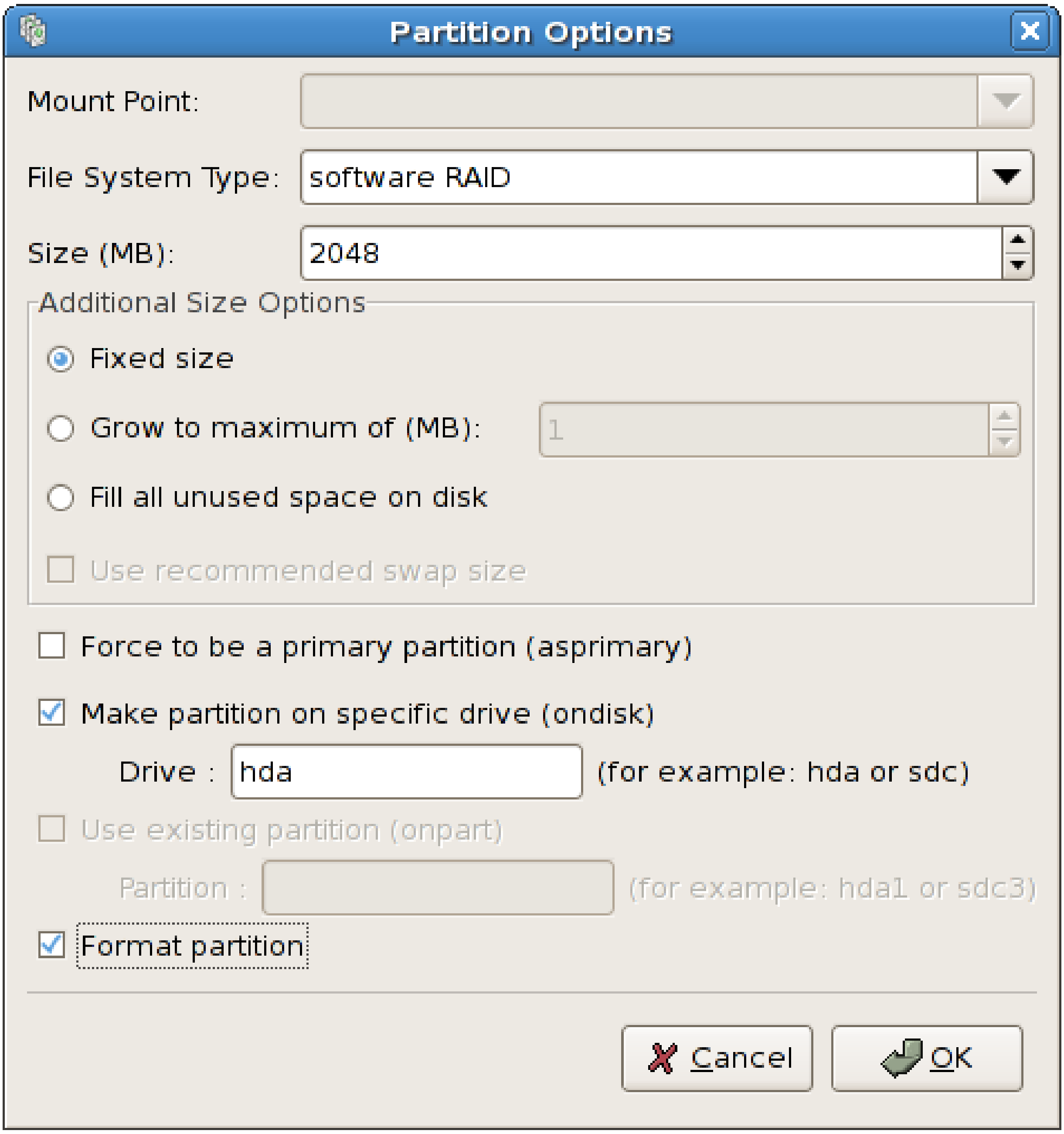

Figure 32.6. Creating a Software RAID Partition

Repeat these steps to create as many partitions as needed for your RAID setup. All of your partitions do not have to be RAID partitions.

After creating all the partitions needed to form a RAID device, follow these steps:

1. Click the **`RAID`** button.
2. Select **`Create a RAID device`**.
3. Select a mount point, file system type, RAID device name, RAID level, RAID members, number of spares for the software RAID device, and whether to format the RAID device.

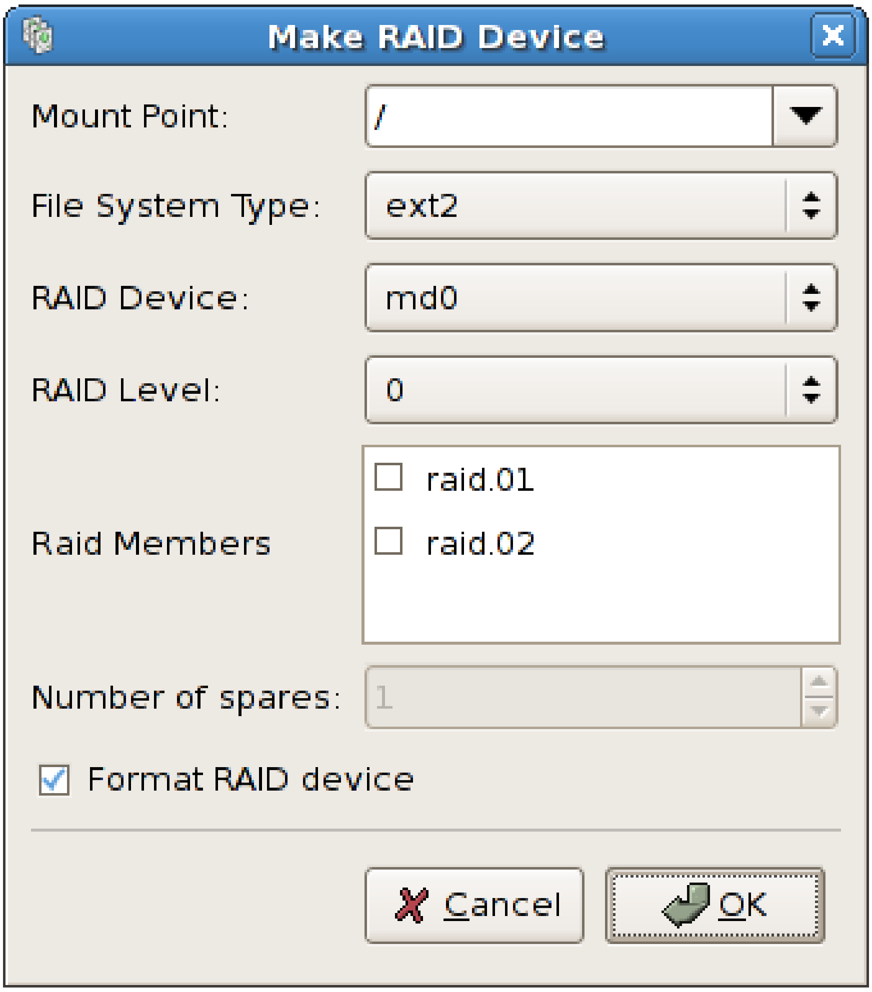

Figure 32.7. Creating a Software RAID Device

4. Click **OK** to add the device to the list.

32.5. Network Configuration

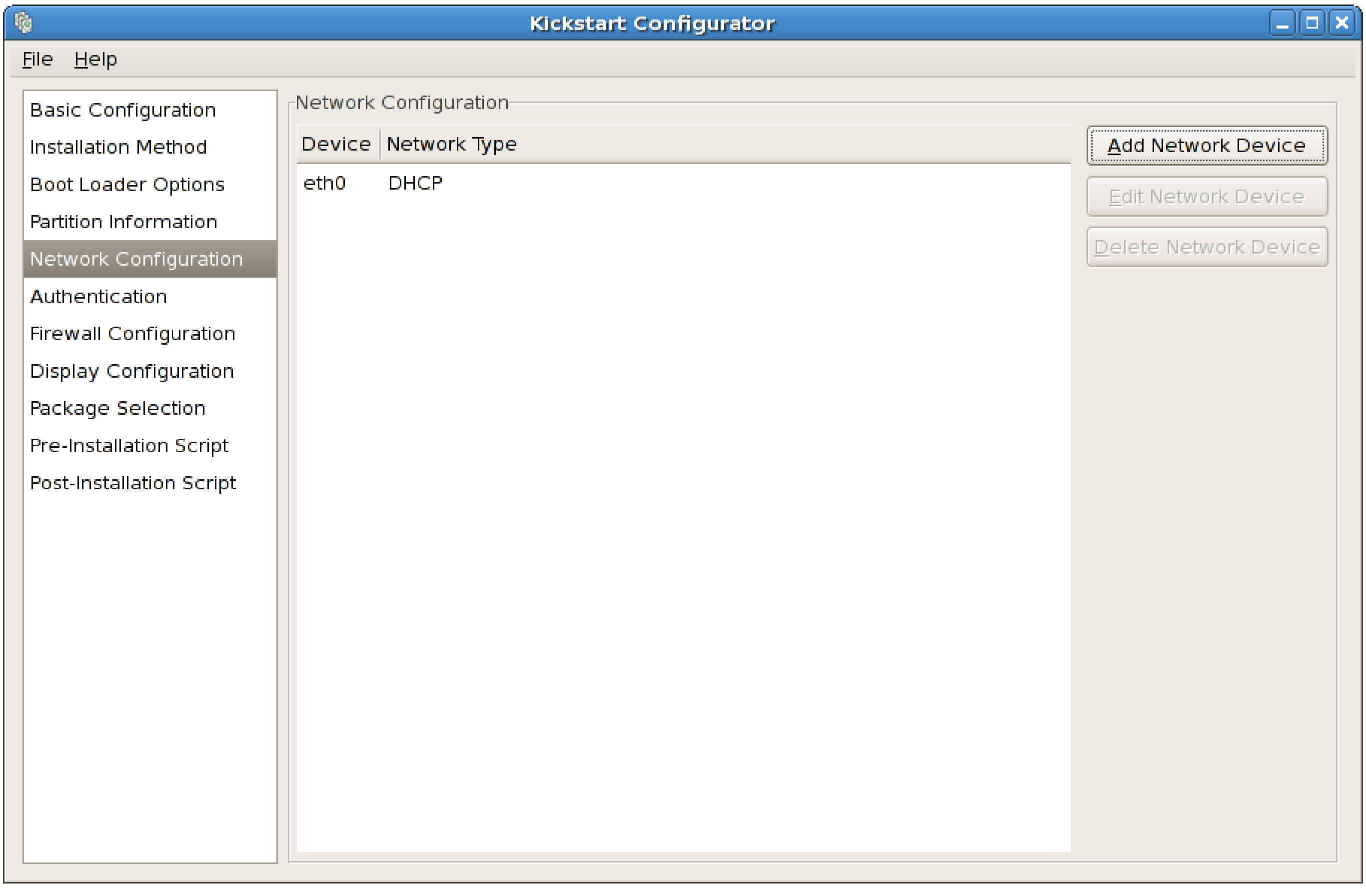

Figure 32.8. Network Configuration

If the system to be installed via kickstart does not have an Ethernet card, do not configure one on the **`Network Configuration`** page.

Networking is only required if you choose a networking-based installation method (NFS, FTP, or HTTP). Networking can always be configured after installation with the **Network Administration Tool** (**`system-config-network`**). Refer to the Red Hat Enterprise Linux Deployment Guide for details.

For each Ethernet card on the system, click **`Add Network Device`** and select the network device and network type for the device. Select **`eth0`** to configure the first Ethernet card, **`eth1`** for the second Ethernet card, and so on.

32.6. Authentication

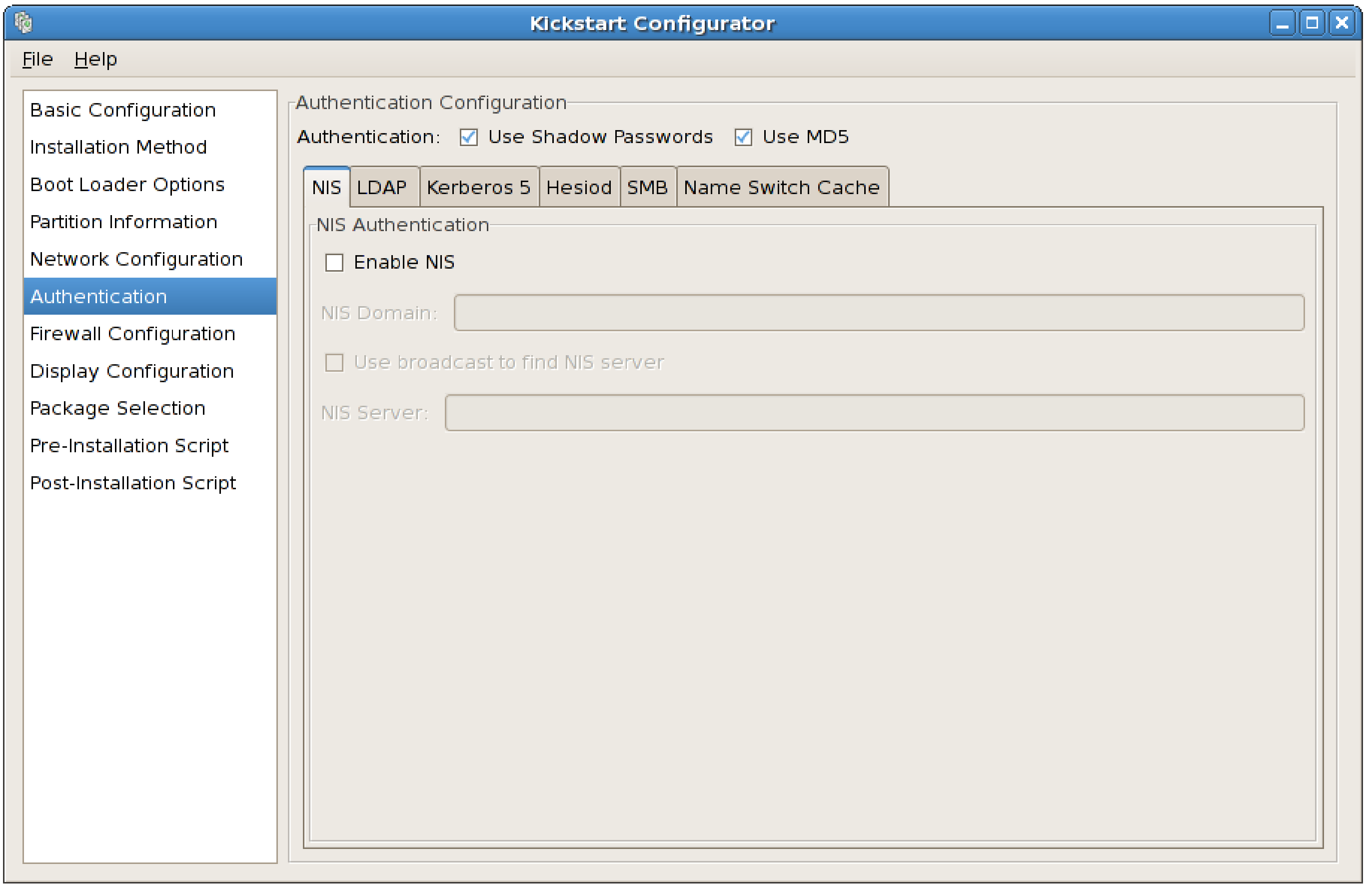

Figure 32.9. Authentication

In the **Authentication** section, select whether to use shadow passwords and MD5 encryption for user passwords. These options are highly recommended and chosen by default.

The **Authentication Configuration** options allow you to configure the following methods of authentication:

- NIS
- LDAP
- Kerberos 5
- Hesiod
- SMB
- Name Switch Cache

These methods are not enabled by default. To enable one or more of these methods, click the appropriate tab, click the checkbox next to **Enable**, and enter the appropriate information for the authentication method. Refer to the Red Hat Enterprise Linux Deployment Guide for more information about the options.

32.7. Firewall Configuration

The **Firewall Configuration** window is similar to the screen in the installation program and the **Security Level Configuration Tool**.

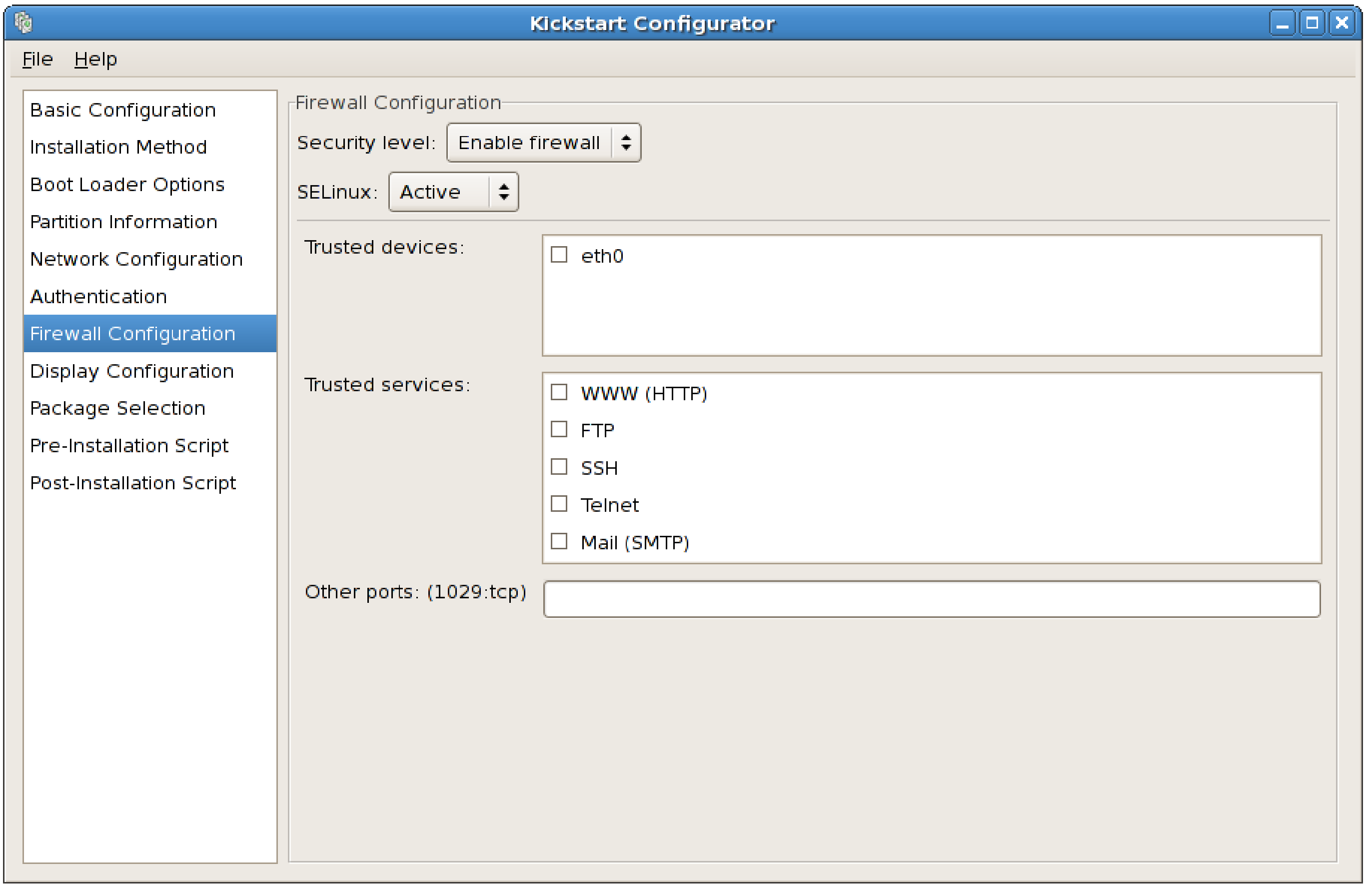

Figure 32.10. Firewall Configuration

If **`Disable firewall`** is selected, the system allows complete access to any active services and ports. No connections to the system are refused or denied.

Selecting **`Enable firewall`** configures the system to reject incoming connections that are not in response to outbound requests, such as DNS replies or DHCP requests. If access to services running on this machine is required, you can choose to allow specific services through the firewall.

Only devices configured in the **`Network Configuration`** section are listed as available **`Trusted devices`**. Connections from any devices selected in the list are accepted by the system. For example, if **`eth1`** only receives connections from internal system, you might want to allow connections from it.

If a service is selected in the **`Trusted services`** list, connections for the service are accepted and processed by the system.

In the **`Other ports`** text field, list any additional ports that should be opened for remote access. Use the following format: **`port:protocol`**. For example, to allow IMAP access through the firewall, specify **`imap:tcp`**. Numeric ports can also be specified explicitly; to allow UDP packets on port 1234 through the firewall, enter **`1234:udp`**. To specify multiple ports, separate them with commas.

32.7.1. SELinux Configuration

Kickstart can set SELinux to **`enforcing`**, **`permissive`** or **`disabled`** mode. Finer grained configuration is not possible at this time.

32.8. Display Configuration

If you are installing the X Window System, you can configure it during the kickstart installation by checking the **`Configure the X Window System`** option on the **`Display Configuration`** window as shown in Figure 32.11, “X Configuration - General”. If this option is not chosen, the X configuration options are disabled and the **`skipx`** option is written to the kickstart file.

32.8.1. General

The first step in configuring X is to choose the default color depth and resolution. Select them from their respective pulldown menus. Be sure to specify a color depth and resolution that is compatible with the video card and monitor for the system.

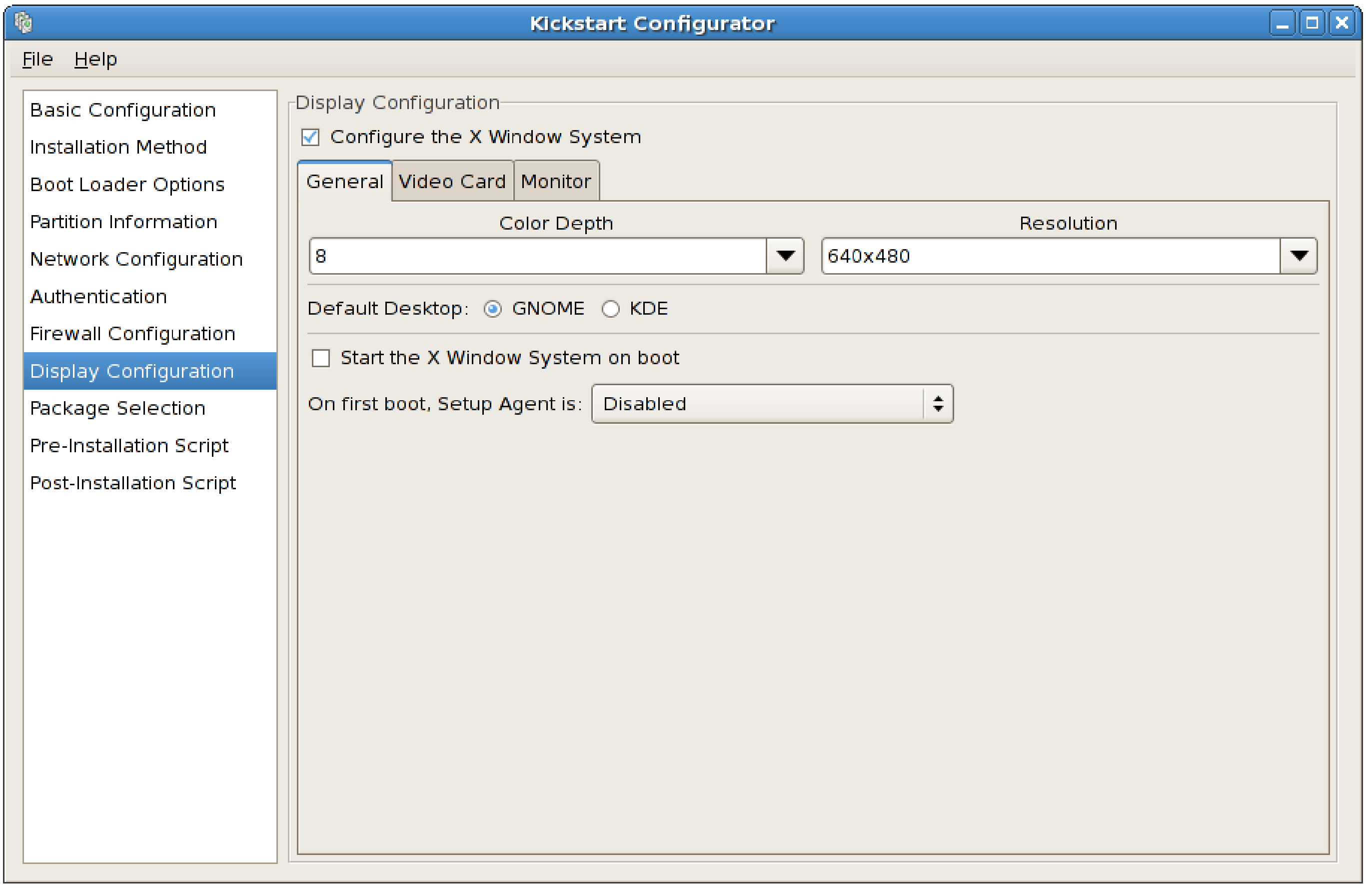

Figure 32.11. X Configuration - General

If you are installing both the GNOME and KDE desktops, you must choose which desktop should be the default. If only one desktop is to be installed, be sure to choose it. Once the system is installed, users can choose which desktop they want to be their default.

Next, choose whether to start the X Window System when the system is booted. This option starts the system in runlevel 5 with the graphical login screen. After the system is installed, this can be changed by modifying the **`/etc/inittab`** configuration file.

Also select whether to start the **Setup Agent** the first time the system is rebooted. It is disabled by default, but the setting can be changed to enabled or enabled in reconfiguration mode. Reconfiguration mode enables the language, mouse, keyboard, root password, security level, time zone, and networking configuration options in addition to the default ones.

32.8.2. Video Card

`Probe for video card driver` is selected by default. Accept this default to have the installation

program probe for the video card during installation. Probing works for most modern video cards. If this option is selected and the installation program cannot successfully probe the video card, the installation program stops at the video card configuration screen. To continue the installation process, select the driver for your video card from the list and click **Next**.

Alternatively, you can select the video card driver from the list on the **Video Card** tab as shown in Figure 32.12, "X Configuration - Video Card". Specify the amount of video RAM the selected video card has from the **Video Card RAM** pulldown menu. These values are used by the installation program to configure the X Window System.

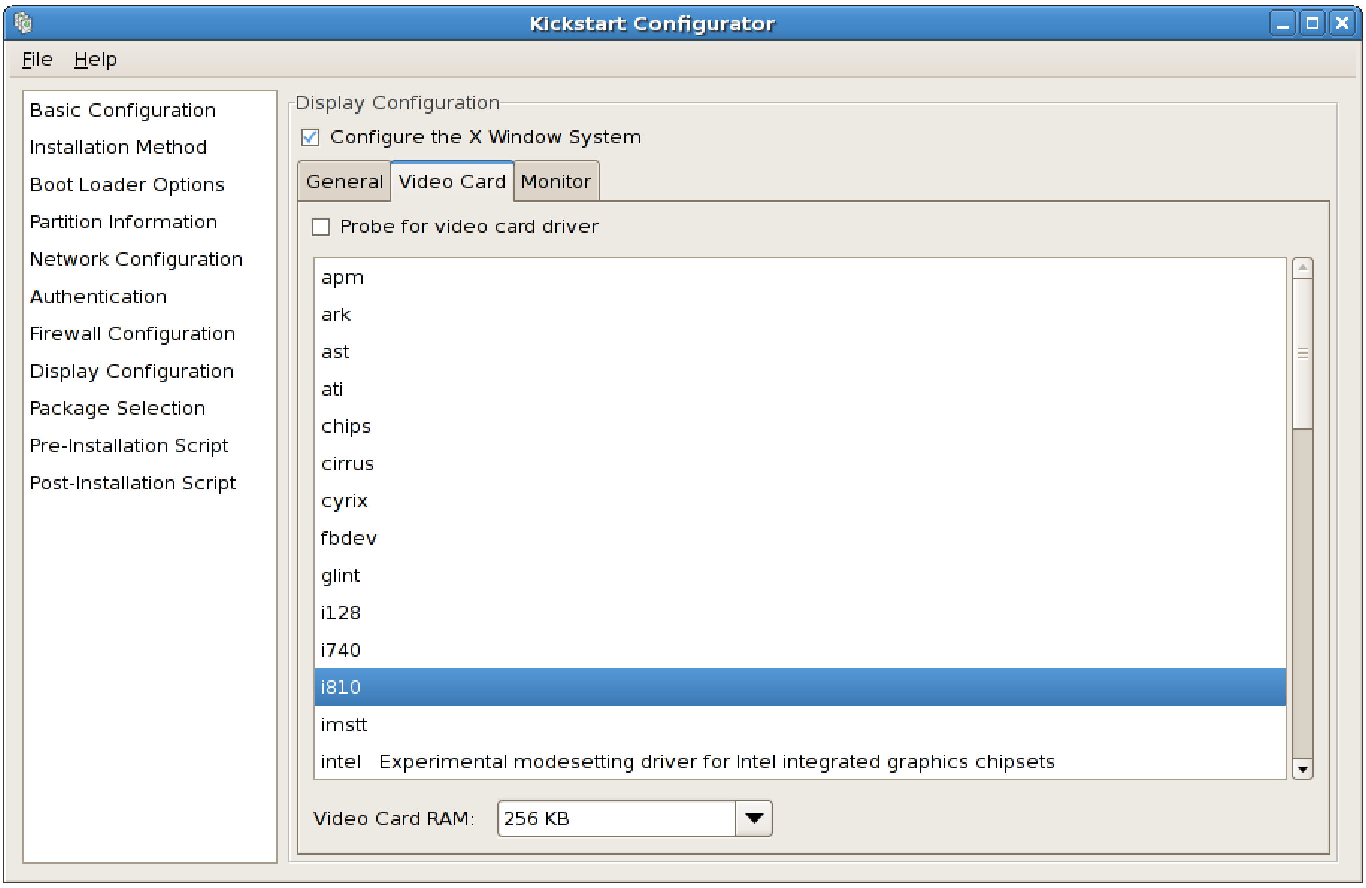

Figure 32.12. X Configuration - Video Card

32.8.3. Monitor

After configuring the video card, click on the **Monitor** tab as shown in Figure 32.13, "X Configuration - Monitor".

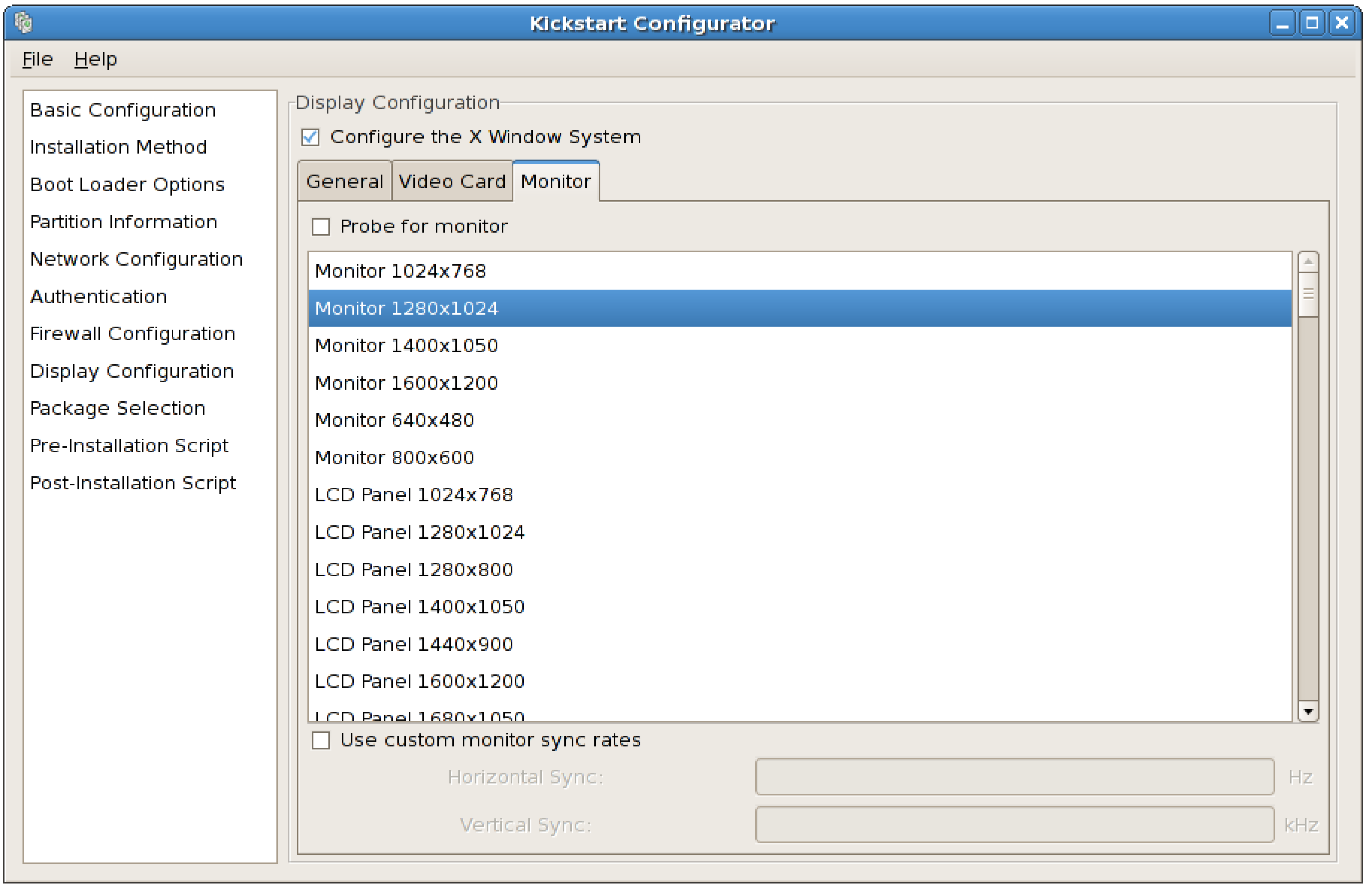

Figure 32.13. X Configuration - Monitor

`Probe for monitor` is selected by default. Accept this default to have the installation program probe for the monitor during installation. Probing works for most modern monitors. If this option is selected and the installation program cannot successfully probe the monitor, the installation program stops at the monitor configuration screen. To continue the installation process, select your monitor from the list and click **Next**.

Alternatively, you can select your monitor from the list. You can also specify the horizontal and vertical sync rates instead of selecting a specific monitor by checking the **`Specify hsync and vsync instead of monitor`** option. This option is useful if the monitor for the system is not listed. Notice that when this option is enabled, the monitor list is disabled.

32.9. Package Selection

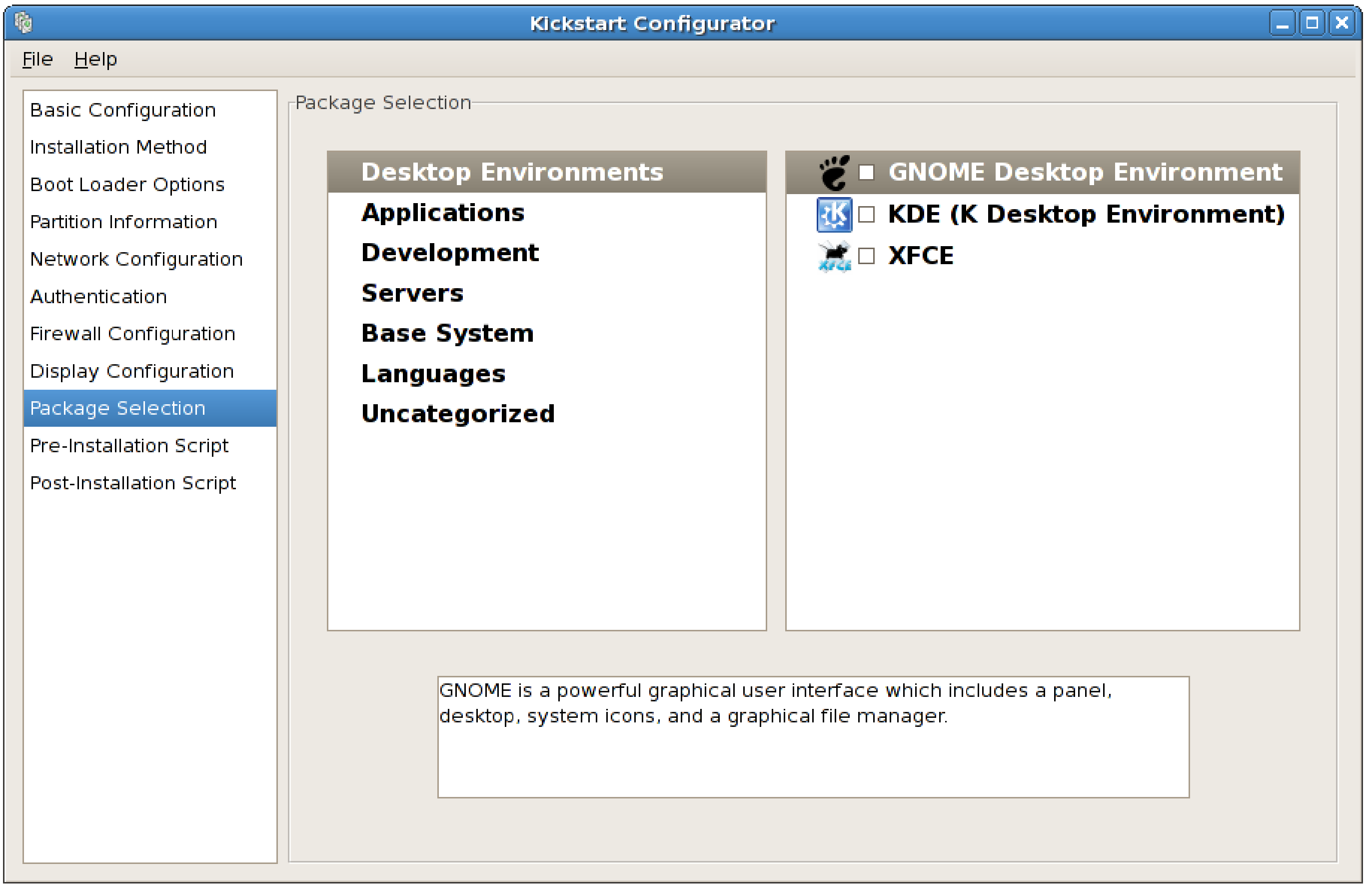

Figure 32.14. Package Selection

The **`Package Selection`** window allows you to choose which package groups to install.

Package resolution is carried out automatically.

Currently, **Kickstart Configurator** does not allow you to select individual packages. To install individual packages, modify the **`%packages`** section of the kickstart file after you save it. Refer to Section 31.5, “Package Selection” for details.

32.10. Pre-Installation Script

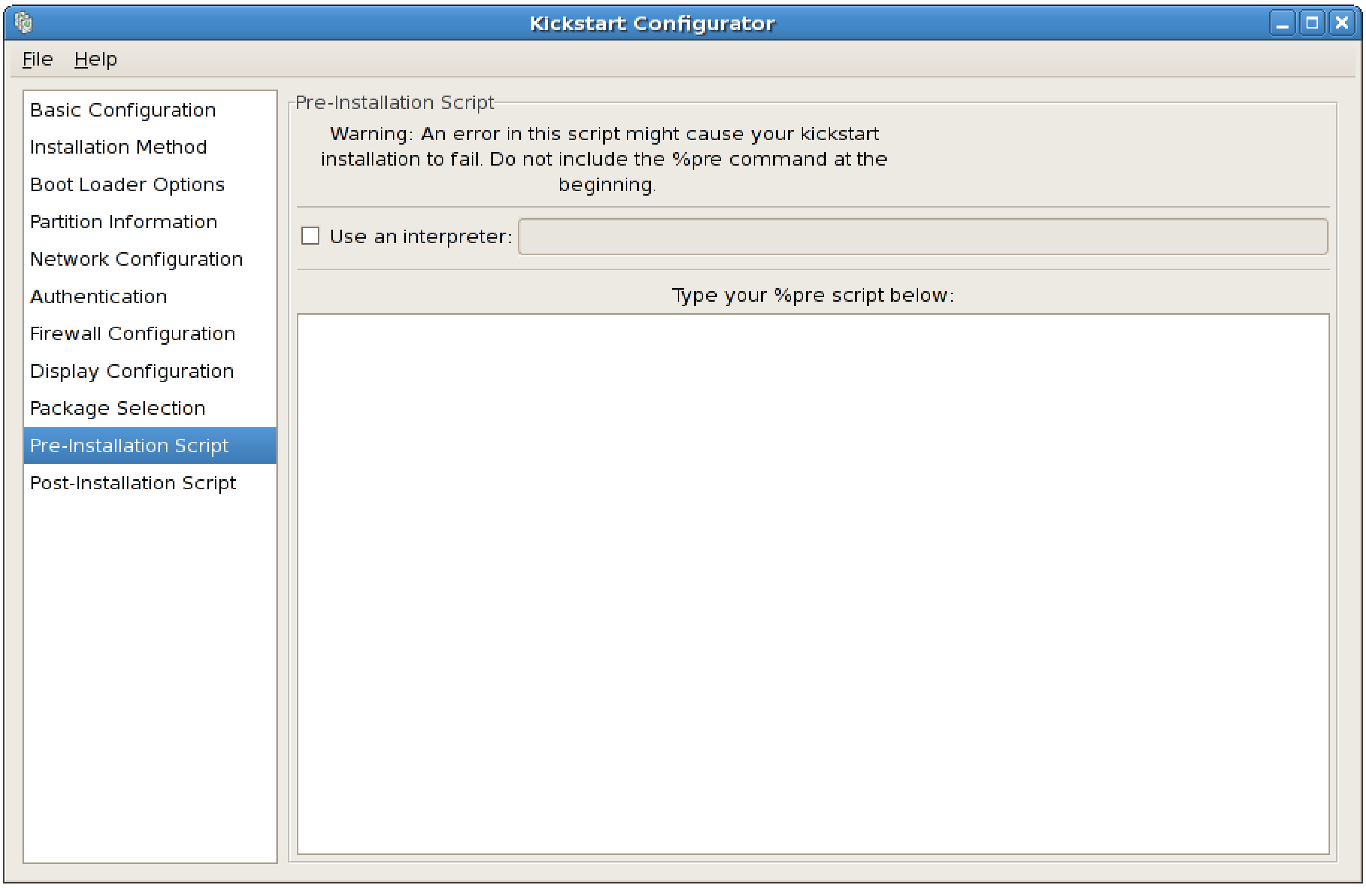

Figure 32.15. Pre-Installation Script

You can add commands to run on the system immediately after the kickstart file has been parsed and before the installation begins. If you have configured the network in the kickstart file, the network is enabled before this section is processed. To include a pre-installation script, type it in the text area.

To specify a scripting language to use to execute the script, select the **`Use an interpreter`** option and enter the interpreter in the text box beside it. For example, **`/usr/bin/python2.4`** can be specified for a Python script. This option corresponds to using **`%pre --interpreter`** ***`/usr/bin/python2.4`*** in your kickstart file.

Many of the commands that are available in the pre-installation environment are provided by a version of **busybox** called **busybox-anaconda**. **Busybox**-supplied commands do not provide all features, but supply only the most commonly used features. The following list of available commands include commands provided by **busybox**:

`addgroup`, **`adduser`**, **`adjtimex`**, **`ar`**, **`arping`**, **`ash`**, **`awk`**, **`basename`**, **`bbconfig`**, **`bunzip2`**, **`busybox`**, **`bzcat`**, **`cal`**, **`cat`**, **`catv`**, **`chattr`**, **`chgrp`**, **`chmod`**, **`chown`**, **`chroot`**, **`chvt`**, **`cksum`**, **`clear`**, **`cmp`**, **`comm`**, **`cp`**, **`cpio`**, **`crond`**, **`crontab`**, **`cut`**, **`date`**, **`dc`**, **`dd`**, **`deallocvt`**, **`delgroup`**, **`deluser`**, **`devfsd`**, **`df`**, **`diff`**, **`dirname`**, **`dmesg`**, **`dnsd`**, **`dos2unix`**, **`dpkg`**, **`dpkg-deb`**, **`du`**, **`dumpkmap`**, **`dumpleases`**, **`e2fsck`**, **`e2label`**, **`echo`**, **`ed`**, **`egrep`**, **`eject`**, **`env`**, **`ether-wake`**, **`expr`**, **`fakeidentd`**, **`false`**, **`fbset`**, **`fdflush`**, **`fdformat`**, **`fdisk`**, **`fgrep`**, **`find`**, **`findfs`**, **`fold`**, **`free`**, **`freeramdisk`**, **`fsck`**, **`fsck.ext2`**, **`fsck.ext3`**, **`fsck.minix`**, **`ftpget`**, **`ftpput`**, **`fuser`**, **`getopt`**, **`getty`**, **`grep`**, **`gunzip`**, **`gzip`**, **`hdparm`**, **`head`**, **`hexdump`**, **`hostid`**, **`hostname`**, **`httpd`**, **`hush`**, **`hwclock`**, **`id`**, **`ifconfig`**, **`ifdown`**, **`ifup`**, **`inetd`**, **`insmod`**, **`install`**, **`ip`**, **`ipaddr`**, **`ipcalc`**, **`ipcrm`**, **`ipcs`**, **`iplink`**, **`iproute`**, **`iptunnel`**, **`kill`**, **`killall`**, **`lash`**, **`last`**, **`length`**, **`less`**, **`linux32`**, **`linux64`**, **`ln`**, **`load_policy`**, **`loadfont`**, **`loadkmap`**, **`login`**, **`logname`**, **`losetup`**, **`ls`**, **`lsattr`**, **`lsmod`**, **`lzmacat`**, **`makedevs`**, **`md5sum`**, **`mdev`**, **`mesg`**, **`mkdir`**, **`mke2fs`**, **`mkfifo`**, **`mkfs.ext2`**, **`mkfs.ext3`**, **`mkfs.minix`**, **`mknod`**, **`mkswap`**, **`mktemp`**, **`modprobe`**, **`more`**, **`mount`**, **`mountpoint`**, **`msh`**, **`mt`**, **`mv`**, **`nameif`**, **`nc`**,

netstat, **nice**, **nohup**, **nslookup**, **od**, **openvt**, **passwd**, **patch**, **pidof**, **ping**, **ping6**, **pipe_progress**, **pivot_root**, **printenv**, **printf**, **ps**, **pwd**, **rdate**, **readlink**, **readprofile**, **realpath**, **renice**, **reset**, **rm**, **rmdir**, **rmmod**, **route**, **rpm**, **rpm2cpio**, **run-parts**, **runlevel**, **rx**, **sed**, **seq**, **setarch**, **setconsole**, **setkeycodes**, **setlogcons**, **setsid**, **sh**, **sha1sum**, **sleep**, **sort**, **start-stop-daemon**, **stat**, **strings**, **stty**, **su**, **sulogin**, **sum**, **swapoff**, **swapon**, **switch_root**, **sync**, **sysctl**, **tail**, **tar**, **tee**, **telnet**, **telnetd**, **test**, **tftp**, **time**, **top**, **touch**, **tr**, **traceroute**, **true**, **tty**, **tune2fs**, **udhcpc**, **udhcpd**, **umount**, **uname**, **uncompress**, **uniq**, **unix2dos**, **unlzma**, **unzip**, **uptime**, **usleep**, **uudecode**, **uuencode**, **vconfig**, **vi**, **vlock**, **watch**, **watchdog**, **wc**, **wget**, **which**, **who**, **whoami**, **xargs**, **yes**, **zcat**, **zcip**

For a description of any of these commands, run:

busybox ***command*** **--help**

In addition to the aforementioned commands, the following commands are provided in their full featured versions:

anaconda, **bash**, **bzip2**, **jmacs**, **ftp**, **head**, **joe**, **kudzu-probe**, **list-harddrives**, **loadkeys**, **mtools**, **mbchk**, **mtools**, **mini-wm**, **mtools**, **jpico**, **pump**, **python**, **python2.4**, **raidstart**, **raidstop**, **rcp**, **rlogin**, **rsync**, **setxkbmap**, **sftp**, **shred**, **ssh**, **syslinux**, **syslogd**, **tac**, **termidx**, **vncconfig**, **vncpasswd**, **xkbcomp**, **Xorg**, **Xvnc**, **zcat**

Warning

Do not include the **%pre** command. It is added for you.

Note

The pre-installation script is run after the source media is mounted and stage 2 of the bootloader has been loaded. For this reason it is not possible to change the source media in the pre-installation script.

32.11. Post-Installation Script

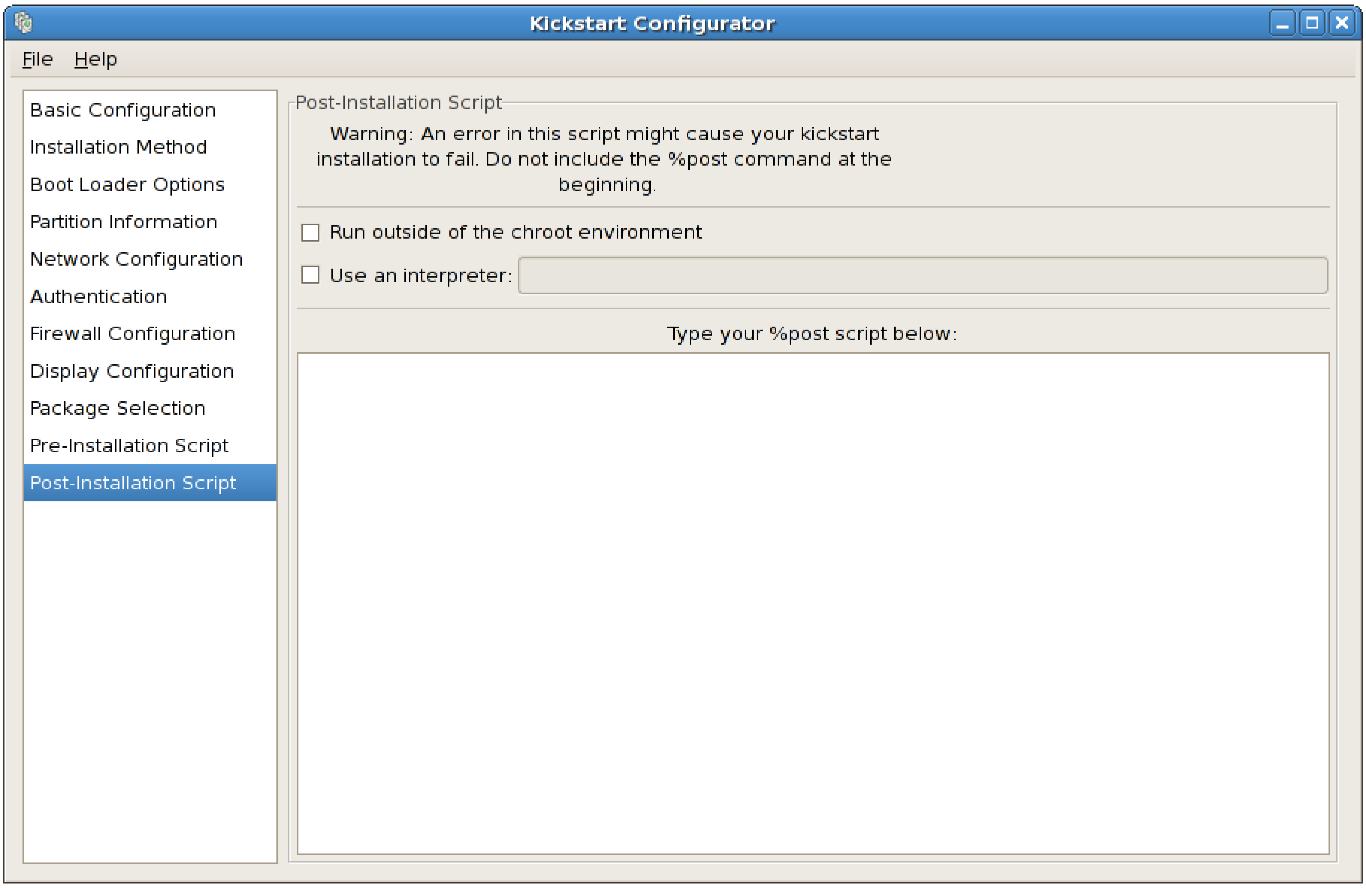

Figure 32.16. Post-Installation Script

You can also add commands to execute on the system after the installation is completed. If the network is properly configured in the kickstart file, the network is enabled, and the script can include commands to access resources on the network. To include a post-installation script, type it in the text area.

> **Warning**
>
> Do not include the **`%post`** command. It is added for you.

For example, to change the message of the day for the newly installed system, add the following command to the **`%post`** section:

```
echo "Hackers will be punished" > /etc/motd
```

> **Note**
>
> More examples can be found in Section 31.7.1, “Examples”.

32.11.1. Chroot Environment

To run the post-installation script outside of the chroot environment, click the checkbox next to this option on the top of the **`Post-Installation`** window. This is equivalent to using the **`--nochroot`** option in the **`%post`** section.

To make changes to the newly installed file system, within the post-installation section, but outside of the chroot environment, you must prepend the directory name with **/mnt/sysimage/**.

For example, if you select **Run outside of the chroot environment**, the previous example must be changed to the following:

```
echo "Hackers will be punished" > /mnt/sysimage/etc/motd
```

32.11.2. Use an Interpreter

To specify a scripting language to use to execute the script, select the **Use an interpreter** option and enter the interpreter in the text box beside it. For example, **/usr/bin/python2.2** can be specified for a Python script. This option corresponds to using **%post --interpreter *****/usr/bin/python2.2*** in your kickstart file.

32.12. Saving the File

To review the contents of the kickstart file after you have finished choosing your kickstart options, select **File** => **Preview** from the pull-down menu.

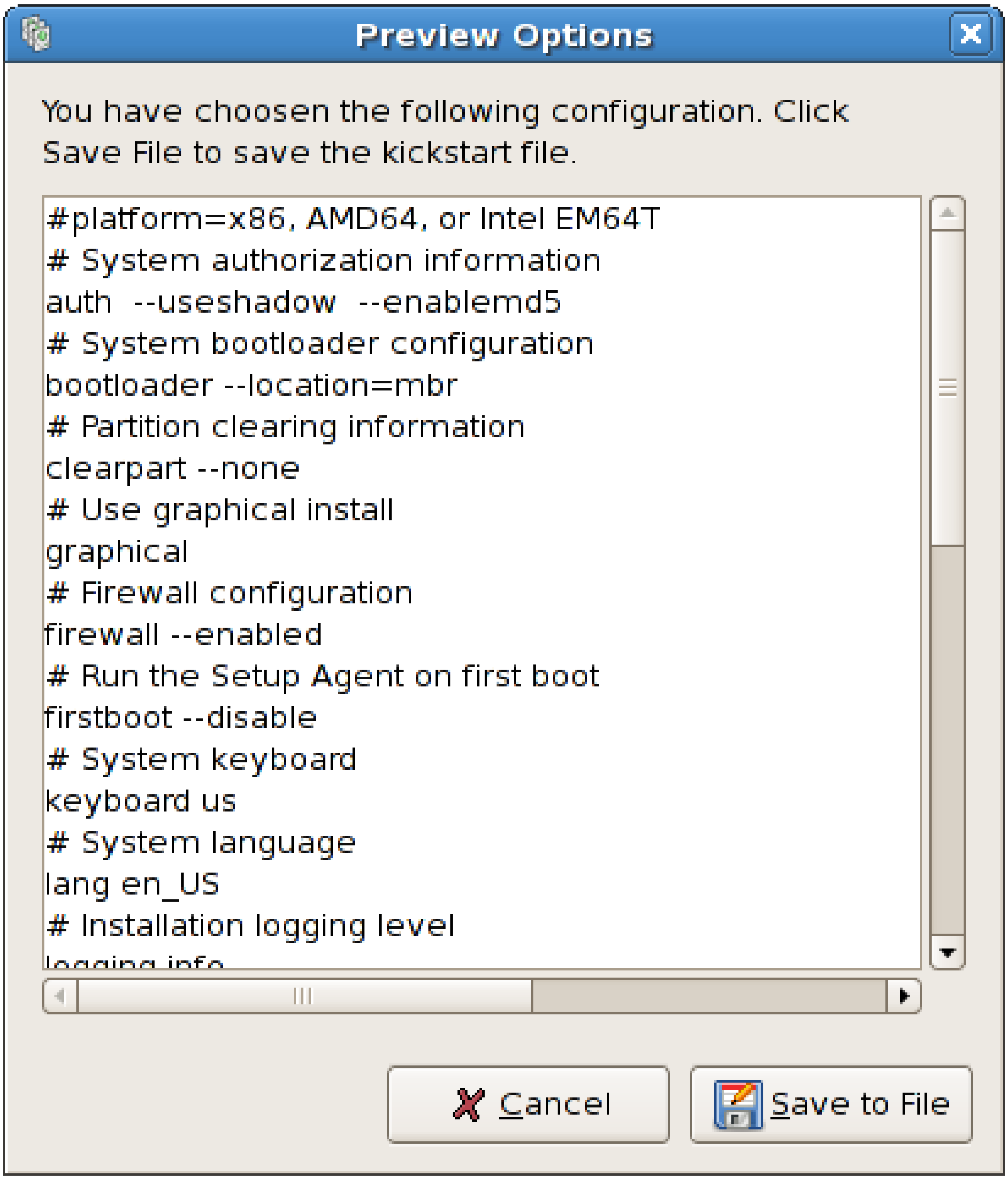

Figure 32.17. Preview

To save the kickstart file, click the **`Save to File`** button in the preview window. To save the file without previewing it, select **File** => **Save File** or press **`Ctrl`**+**`S`** . A dialog box appears. Select where to save the file.

After saving the file, refer to Section 31.10, “Starting a Kickstart Installation” for information on how to start the kickstart installation.

Chapter 33. Boot Process, Init, and Shutdown

An important and powerful aspect of Red Hat Enterprise Linux is the open, user-configurable method it uses for starting the operating system. Users are free to configure many aspects of the boot process, including specifying the programs launched at boot-time. Similarly, system shutdown gracefully terminates processes in an organized and configurable way, although customization of this process is rarely required.

Understanding how the boot and shutdown processes work not only allows customization, but also makes it easier to troubleshoot problems related to starting or shutting down the system.

> **Important — Boot Time Message Logging**
>
> In previous versions of Red Hat Enterprise Linux, boot time messages from init scripts were logged to **`/var/log/boot.log`**. This functionality is not available in Red Hat Enterprise Linux 5.

33.1. The Boot Process

Below are the basic stages of the boot process for an x86 system:

1. The system BIOS checks the system and launches the first stage boot loader on the MBR of the primary hard disk.
2. The first stage boot loader loads itself into memory and launches the second stage boot loader from the **`/boot/`** partition.
3. The second stage boot loader loads the kernel into memory, which in turn loads any necessary modules and mounts the root partition read-only.
4. The kernel transfers control of the boot process to the **`/sbin/init`** program.
5. The **`/sbin/init`** program loads all services and user-space tools, and mounts all partitions listed in **`/etc/fstab`**.
6. The user is presented with a login screen for the freshly booted Linux system.

Because configuration of the boot process is more common than the customization of the shutdown process, the remainder of this chapter discusses in detail how the boot process works and how it can be customized to suite specific needs.

33.2. A Detailed Look at the Boot Process

The beginning of the boot process varies depending on the hardware platform being used. However, once the kernel is found and loaded by the boot loader, the default boot process is identical across all architectures. This chapter focuses primarily on the x86 architecture.

33.2.1. The BIOS

When an x86 computer is booted, the processor looks at the end of system memory for the *Basic Input/Output System* or *BIOS* program and runs it. The BIOS controls not only the first step of the boot process, but also provides the lowest level interface to peripheral devices. For this reason it is written into read-only, permanent memory and is always available for use.

Other platforms use different programs to perform low-level tasks roughly equivalent to those of the BIOS on an x86 system. For instance, Itanium-based computers use the *Extensible Firmware Interface* (*EFI*) *Shell*.

Once loaded, the BIOS tests the system, looks for and checks peripherals, and then locates a valid device with which to boot the system. Usually, it checks any diskette drives and CD-ROM drives present for bootable media, then, failing that, looks to the system's hard drives. In most cases, the order of the drives searched while booting is controlled with a setting in the BIOS, and it looks on the master IDE device on the primary IDE bus. The BIOS then loads into memory whatever program is residing in the first sector of this device, called the *Master Boot Record* or *MBR*. The MBR is only 512 bytes in size and contains machine code instructions for booting the machine, called a boot loader, along with the partition table. Once the BIOS finds and loads the boot loader program into memory, it yields control of the boot process to it.

33.2.2. The Boot Loader

This section looks at the default boot loader for the x86 platform, GRUB. Depending on the system's architecture, the boot process may differ slightly. Refer to Section 33.2.2.1, “Boot Loaders for Other Architectures” for a brief overview of non-x86 boot loaders. For more information about configuring and using GRUB, see Chapter 9, *The GRUB Boot Loader*.

A boot loader for the x86 platform is broken into at least two stages. The first stage is a small machine code binary on the MBR. Its sole job is to locate the second stage boot loader and load the first part of it into memory.

GRUB has the advantage of being able to read ext2 and ext3 [13] partitions and load its configuration file — **`/boot/grub/grub.conf`** — at boot time. Refer to Section 9.7, “GRUB Menu Configuration File” for information on how to edit this file.

Note

If upgrading the kernel using the **Red Hat Update Agent**, the boot loader configuration file is updated automatically. More information on Red Hat Network can be found online at the following URL: https://rhn.redhat.com/.

Once the second stage boot loader is in memory, it presents the user with a graphical screen showing the different operating systems or kernels it has been configured to boot. On this screen a user can use the arrow keys to choose which operating system or kernel they wish to boot and press **`Enter`**. If no key is pressed, the boot loader loads the default selection after a configurable period of time has passed.

Once the second stage boot loader has determined which kernel to boot, it locates the corresponding kernel binary in the **`/boot/`** directory. The kernel binary is named using the following format — **`/boot/vmlinuz-<kernel-version>`** file (where **`<kernel-version>`** corresponds to the kernel version specified in the boot loader's settings).

For instructions on using the boot loader to supply command line arguments to the kernel, refer to Chapter 9, *The GRUB Boot Loader*. For information on changing the runlevel at the boot loader prompt, refer Section 9.8, “Changing Runlevels at Boot Time”.

The boot loader then places one or more appropriate *initramfs* images into memory. Next, the kernel decompresses these images from memory to **/sysroot/**, a RAM-based virtual file system, via **cpio**. The **initramfs** is used by the kernel to load drivers and modules necessary to boot the system. This is particularly important if SCSI hard drives are present or if the systems use the ext3 file system.

Once the kernel and the **initramfs** image(s) are loaded into memory, the boot loader hands control of the boot process to the kernel.

For a more detailed overview of the GRUB boot loader, refer to Chapter 9, *The GRUB Boot Loader*.

33.2.2.1. Boot Loaders for Other Architectures

Once the kernel loads and hands off the boot process to the **init** command, the same sequence of events occurs on every architecture. So the main difference between each architecture's boot process is in the application used to find and load the kernel.

For example, the Itanium architecture uses the ELILO boot loader, the IBM eServer pSeries architecture uses yaboot, and the IBM System z systems use the z/IPL boot loader.

33.2.3. The Kernel

When the kernel is loaded, it immediately initializes and configures the computer's memory and configures the various hardware attached to the system, including all processors, I/O subsystems, and storage devices. It then looks for the compressed **initramfs** image(s) in a predetermined location in memory, decompresses it directly to **/sysroot/**, and loads all necessary drivers. Next, it initializes virtual devices related to the file system, such as LVM or software RAID, before completing the **initramfs** processes and freeing up all the memory the disk image once occupied.

The kernel then creates a root device, mounts the root partition read-only, and frees any unused memory.

At this point, the kernel is loaded into memory and operational. However, since there are no user applications that allow meaningful input to the system, not much can be done with the system.

To set up the user environment, the kernel executes the **/sbin/init** program.

33.2.4. The /sbin/init Program

The **/sbin/init** program (also called **init**) coordinates the rest of the boot process and configures the environment for the user.

When the **init** command starts, it becomes the parent or grandparent of all of the processes that start up automatically on the system. First, it runs the **/etc/rc.d/rc.sysinit** script, which sets the environment path, starts swap, checks the file systems, and executes all other steps required for system initialization. For example, most systems use a clock, so **rc.sysinit** reads the **/etc/sysconfig/clock** configuration file to initialize the hardware clock. Another example is if there are special serial port processes which must be initialized, **rc.sysinit** executes the **/etc/rc.serial** file.

The **init** command then runs the **/etc/inittab** script, which describes how the system should be set up in each *SysV init runlevel*. Runlevels are a state, or *mode*, defined by the services listed in the SysV **/etc/rc.d/rc*<x>*.d/** directory, where *<x>* is the number of the runlevel. For more information on SysV init runlevels, refer to Section 33.4, “SysV Init Runlevels”.

Next, the **init** command sets the source function library, **/etc/rc.d/init.d/functions**, for the system, which configures how to start, kill, and determine the PID of a program.

The **init** program starts all of the background processes by looking in the appropriate **rc** directory for the runlevel specified as the default in **/etc/inittab**. The **rc** directories are numbered to correspond to the runlevel they represent. For instance, **/etc/rc.d/rc5.d/** is the directory for runlevel 5.

When booting to runlevel 5, the **init** program looks in the **/etc/rc.d/rc5.d/** directory to determine which processes to start and stop.

Below is an example listing of the **/etc/rc.d/rc5.d/** directory:

```
K05innd -> ../init.d/innd
K05saslauthd -> ../init.d/saslauthd
K10dc_server -> ../init.d/dc_server
K10psacct -> ../init.d/psacct
K10radiusd -> ../init.d/radiusd
K12dc_client -> ../init.d/dc_client
K12FreeWnn -> ../init.d/FreeWnn
K12mailman -> ../init.d/mailman
K12mysqld -> ../init.d/mysqld
K15httpd -> ../init.d/httpd
K20netdump-server -> ../init.d/netdump-server
K20rstatd -> ../init.d/rstatd
K20rusersd -> ../init.d/rusersd
K20rwhod -> ../init.d/rwhod
K24irda -> ../init.d/irda
K25squid -> ../init.d/squid
K28amd -> ../init.d/amd
K30spamassassin -> ../init.d/spamassassin
K34dhcrelay -> ../init.d/dhcrelay
K34yppasswdd -> ../init.d/yppasswdd
K35dhcpd -> ../init.d/dhcpd
K35smb -> ../init.d/smb
K35vncserver -> ../init.d/vncserver
K36lisa -> ../init.d/lisa
K45arpwatch -> ../init.d/arpwatch
K45named -> ../init.d/named
K46radvd -> ../init.d/radvd
K50netdump -> ../init.d/netdump
K50snmpd -> ../init.d/snmpd
K50snmptrapd -> ../init.d/snmptrapd
K50tux -> ../init.d/tux
K50vsftpd -> ../init.d/vsftpd
K54dovecot -> ../init.d/dovecot
K61ldap -> ../init.d/ldap
K65kadmin -> ../init.d/kadmin
K65kprop -> ../init.d/kprop
K65krb524 -> ../init.d/krb524
K65krb5kdc -> ../init.d/krb5kdc
K70aep1000 -> ../init.d/aep1000
K70bcm5820 -> ../init.d/bcm5820
K74ypserv -> ../init.d/ypserv
K74ypxfrd -> ../init.d/ypxfrd
K85mdmpd -> ../init.d/mdmpd
K89netplugd -> ../init.d/netplugd
K99microcode_ctl -> ../init.d/microcode_ctl
S04readahead_early -> ../init.d/readahead_early
S05kudzu -> ../init.d/kudzu
S06cpuspeed -> ../init.d/cpuspeed
S08ip6tables -> ../init.d/ip6tables
S08iptables -> ../init.d/iptables
S09isdn -> ../init.d/isdn
S10network -> ../init.d/network
S12syslog -> ../init.d/syslog
S13irqbalance -> ../init.d/irqbalance
```

```
S13portmap -> ../init.d/portmap
S15mdmonitor -> ../init.d/mdmonitor
S15zebra -> ../init.d/zebra
S16bgpd -> ../init.d/bgpd
S16ospf6d -> ../init.d/ospf6d
S16ospfd -> ../init.d/ospfd
S16ripd -> ../init.d/ripd
S16ripngd -> ../init.d/ripngd
S20random -> ../init.d/random
S24pcmcia -> ../init.d/pcmcia
S25netfs -> ../init.d/netfs
S26apmd -> ../init.d/apmd
S27ypbind -> ../init.d/ypbind
S28autofs -> ../init.d/autofs
S40smartd -> ../init.d/smartd
S44acpid -> ../init.d/acpid
S54hpoj -> ../init.d/hpoj
S55cups -> ../init.d/cups
S55sshd -> ../init.d/sshd
S56rawdevices -> ../init.d/rawdevices
S56xinetd -> ../init.d/xinetd
S58ntpd -> ../init.d/ntpd
S75postgresql -> ../init.d/postgresql
S80sendmail -> ../init.d/sendmail
S85gpm -> ../init.d/gpm
S87iiim -> ../init.d/iiim
S90canna -> ../init.d/canna
S90crond -> ../init.d/crond
S90xfs -> ../init.d/xfs
S95atd -> ../init.d/atd
S96readahead -> ../init.d/readahead
S97messagebus -> ../init.d/messagebus
S97rhnsd -> ../init.d/rhnsd
S99local -> ../rc.local
```

As illustrated in this listing, none of the scripts that actually start and stop the services are located in the **/etc/rc.d/rc5.d/** directory. Rather, all of the files in **/etc/rc.d/rc5.d/** are *symbolic links* pointing to scripts located in the **/etc/rc.d/init.d/** directory. Symbolic links are used in each of the **rc** directories so that the runlevels can be reconfigured by creating, modifying, and deleting the symbolic links without affecting the actual scripts they reference.

The name of each symbolic link begins with either a **K** or an **S**. The **K** links are processes that are killed on that runlevel, while those beginning with an **S** are started.

The **init** command first stops all of the **K** symbolic links in the directory by issuing the **/etc/rc.d/init.d/<*command*> stop** command, where <*command*> is the process to be killed. It then starts all of the **S** symbolic links by issuing **/etc/rc.d/init.d/<*command*> start**.

Note

After the system is finished booting, it is possible to log in as root and execute these same scripts to start and stop services. For instance, the command **/etc/rc.d/init.d/httpd stop** stops the Apache HTTP Server.

Each of the symbolic links are numbered to dictate start order. The order in which the services are started or stopped can be altered by changing this number. The lower the number, the earlier it is started. Symbolic links with the same number are started alphabetically.

> **Note**
>
> One of the last things the **init** program executes is the **/etc/rc.d/rc.local** file. This file is useful for system customization. Refer to Section 33.3, "Running Additional Programs at Boot Time" for more information about using the **rc.local** file.

After the **init** command has progressed through the appropriate **rc** directory for the runlevel, the **/etc/inittab** script forks an **/sbin/mingetty** process for each virtual console (login prompt) allocated to the runlevel. Runlevels 2 through 5 have all six virtual consoles, while runlevel 1 (single user mode) has one, and runlevels 0 and 6 have none. The **/sbin/mingetty** process opens communication pathways to *tty* devices [14], sets their modes, prints the login prompt, accepts the user's username and password, and initiates the login process.

In runlevel 5, the **/etc/inittab** runs a script called **/etc/X11/prefdm**. The **prefdm** script executes the preferred X display manager [15] — **gdm**, **kdm**, or **xdm**, depending on the contents of the **/etc/sysconfig/desktop** file.

Once finished, the system operates on runlevel 5 and displays a login screen.

33.3. Running Additional Programs at Boot Time

The **/etc/rc.d/rc.local** script is executed by the **init** command at boot time or when changing runlevels. Adding commands to the bottom of this script is an easy way to perform necessary tasks like starting special services or initialize devices without writing complex initialization scripts in the **/etc/rc.d/init.d/** directory and creating symbolic links.

The **/etc/rc.serial** script is used if serial ports must be setup at boot time. This script runs **setserial** commands to configure the system's serial ports. Refer to the **setserial** man page for more information.

33.4. SysV Init Runlevels

The SysV init runlevel system provides a standard process for controlling which programs **init** launches or halts when initializing a runlevel. SysV init was chosen because it is easier to use and more flexible than the traditional BSD-style init process.

The configuration files for SysV init are located in the **/etc/rc.d/** directory. Within this directory, are the **rc**, **rc.local**, **rc.sysinit**, and, optionally, the **rc.serial** scripts as well as the following directories:

```
init.d/ rc0.d/ rc1.d/ rc2.d/ rc3.d/ rc4.d/ rc5.d/ rc6.d/
```

The **init.d/** directory contains the scripts used by the **/sbin/init** command when controlling services. Each of the numbered directories represent the six runlevels configured by default under Red Hat Enterprise Linux.

33.4.1. Runlevels

The idea behind SysV init runlevels revolves around the idea that different systems can be used in different ways. For example, a server runs more efficiently without the drag on system resources created by the X Window System. Or there may be times when a system administrator may need to

operate the system at a lower runlevel to perform diagnostic tasks, like fixing disk corruption in runlevel 1.

The characteristics of a given runlevel determine which services are halted and started by **`init`**. For instance, runlevel 1 (single user mode) halts any network services, while runlevel 3 starts these services. By assigning specific services to be halted or started on a given runlevel, **`init`** can quickly change the mode of the machine without the user manually stopping and starting services.

The following runlevels are defined by default under Red Hat Enterprise Linux:

- **`0`** — Halt
- **`1`** — Single-user text mode
- **`2`** — Not used (user-definable)
- **`3`** — Full multi-user text mode
- **`4`** — Not used (user-definable)
- **`5`** — Full multi-user graphical mode (with an X-based login screen)
- **`6`** — Reboot

In general, users operate Red Hat Enterprise Linux at runlevel 3 or runlevel 5 — both full multi-user modes. Users sometimes customize runlevels 2 and 4 to meet specific needs, since they are not used.

The default runlevel for the system is listed in **`/etc/inittab`**. To find out the default runlevel for a system, look for the line similar to the following near the top of **`/etc/inittab`**:

```
id:5:initdefault:
```

The default runlevel listed in this example is five, as the number after the first colon indicates. To change it, edit **`/etc/inittab`** as root.

Warning

Be very careful when editing **`/etc/inittab`**. Simple typos can cause the system to become unbootable. If this happens, either use a boot diskette, enter single-user mode, or enter rescue mode to boot the computer and repair the file.

For more information on single-user and rescue mode, see the chapter titled *Basic System Recovery* in the Red Hat Enterprise Linux Deployment Guide.

It is possible to change the default runlevel at boot time by modifying the arguments passed by the boot loader to the kernel. For information on changing the runlevel at boot time, refer to Section 9.8, “Changing Runlevels at Boot Time”.

33.4.2. Runlevel Utilities

One of the best ways to configure runlevels is to use an *initscript utility*. These tools are designed to simplify the task of maintaining files in the SysV init directory hierarchy and relieves system administrators from having to directly manipulate the numerous symbolic links in the subdirectories of **`/etc/rc.d/`**.

Red Hat Enterprise Linux provides three such utilities:

- **/sbin/chkconfig** — The **/sbin/chkconfig** utility is a simple command line tool for maintaining the **/etc/rc.d/init.d/** directory hierarchy.
- **/usr/sbin/ntsysv** — The ncurses-based **/sbin/ntsysv** utility provides an interactive text-based interface, which some find easier to use than **chkconfig**.
- **Services Configuration Tool** — The graphical **Services Configuration Tool** (**system-config-services**) program is a flexible utility for configuring runlevels.

See the chapter titled *Controlling Access to Services* in the Red Hat Enterprise Linux Deployment Guide for more information regarding these tools.

33.5. Shutting Down

To shut down Red Hat Enterprise Linux, the root user may issue the **/sbin/shutdown** command. The **shutdown** man page has a complete list of options, but the two most common uses are:

```
/sbin/shutdown -h now
```

and

```
/sbin/shutdown -r now
```

After shutting everything down, the **-h** option halts the machine, and the **-r** option reboots.

PAM console users can use the **reboot** and **halt** commands to shut down the system while in runlevels 1 through 5. For more information about PAM console users, refer to the Red Hat Enterprise Linux Deployment Guide.

If the computer does not power itself down, be careful not to turn off the computer until a message appears indicating that the system is halted.

Failure to wait for this message can mean that not all the hard drive partitions are unmounted, which can lead to file system corruption.

[13] GRUB reads ext3 file systems as ext2, disregarding the journal file. See the chapter titled *The ext3 File System* in the Red Hat Enterprise Linux Deployment Guide for more information on the ext3 file system.

[14] Refer to the Red Hat Enterprise Linux Deployment Guide for more information about **tty** devices.

[15] Refer to the Red Hat Enterprise Linux Deployment Guide for more information about display managers.

Chapter 34. PXE Network Installations

Red Hat Enterprise Linux allows for installation over a network using the NFS, FTP, or HTTP protocols. A network installation can be started from a boot CD-ROM, a bootable flash memory drive, or by using the **askmethod** boot option with the Red Hat Enterprise Linux CD #1. Alternatively, if the system to be installed contains a network interface card (NIC) with Pre-Execution Environment (PXE) support, it can be configured to boot from files on another networked system rather than local media such as a CD-ROM.

For a PXE network installation, the client's NIC with PXE support sends out a broadcast request for DHCP information. The DHCP server provides the client with an IP address, other network information such as name server, the IP address or hostname of the **tftp** server (which provides the files necessary to start the installation program), and the location of the files on the **tftp** server. This is possible because of PXELINUX, which is part of the **syslinux** package.

The following steps must be performed to prepare for a PXE installation:

1. Configure the network (NFS, FTP, HTTP) server to export the installation tree.
2. Configure the files on the **tftp** server necessary for PXE booting.
3. Configure which hosts are allowed to boot from the PXE configuration.
4. Start the **tftp** service.
5. Configure DHCP.
6. Boot the client, and start the installation.

34.1. Setting up the Network Server

First, configure an NFS, FTP, or HTTP server to export the entire installation tree for the version and variant of Red Hat Enterprise Linux to be installed. Refer to the section *Preparing for a Network Installation* in the *Red Hat Enterprise Linux Installation Guide* for detailed instructions.

34.2. PXE Boot Configuration

The next step is to copy the files necessary to start the installation to the **tftp** server so they can be found when the client requests them. The **tftp** server is usually the same server as the network server exporting the installation tree.

To copy these files, run the **Network Booting Tool** on the NFS, FTP, or HTTP server. A separate PXE server is not necessary.

34.2.1. Command Line Configuration

If the network server is not running X, the **pxeos** command line utility, which is part of the **system-config-netboot-cmd** package, can be used to configure the **tftp** server files as described in Section 34.4, “TFTPD”:

```
pxeos -a -i "<description>" -p <NFS|HTTP|FTP> -D 0 -s installer.example.com \
-L <location> -k <kernel> -K <kickstart> <os-identifer>
```

The following list explains the options:

- **-a** — Specifies that an OS instance is being added to the PXE configuration.
- **-i** "*<description>*" — Replace "*<description>*" with a description of the OS instance.
- **-p** <NFS|HTTP|FTP> — Specify which of the NFS, FTP, or HTTP protocols to use for installation. Only one may be specified.
- **-D** <0|1> — Specify "**0**" which indicates that it is *not* a diskless configuration since **pxeos** can be used to configure a diskless environment as well.
- **-s** *installer.example.com* — Provide the name of the NFS, FTP, or HTTP server after the **-s** option.
- **-L** *<location>* — Provide the location of the installation tree on that server after the **-L** option.

  For example, if the installation tree is exported as **/install/rhel5** on an NFS share, specify **-L /install/rhel5**.
- **-k** *<kernel>* — Provide the specific kernel for booting. Installation trees can contain multiple kernels.

  For example, if the installation tree contain a patched kernel named **vmlinuz-du** alongside the standard kernel named **vmlinuz**, use **-k vmlinuz-du** to specify the patched kernel.
- **-K** *<kickstart>* — Provide the location of the kickstart file, if available. Specify this location as a full path, including the protocol; for example: **-K nfs:192.168.0.1:/install/rhel5/ks.cfg**
- *<os-identifer>* — Specify the OS identifier, which is used as the directory name in the **/tftpboot/linux-install/** directory.

If FTP is selected as the installation protocol and anonymous login is not available, specify a username and password for login, with the following options before *<os-identifer>* in the previous command:

```
-A 0 -u <username> -p <password>
```

pxeos writes the results to the **/tftpboot/linux-install/pxelinux.cfg/pxeos.xml** file.

For more information on command line options available for the **pxeos** command, refer to the **pxeos** man page.

The **pxeboot** tool can also edit the **/tftpboot/linux-install/pxelinux.cfg/pxeos.xml** file and uses similar options to **pxeos**. Refer to the **pxeboot** man page for more detail.

34.3. Adding PXE Hosts

After configuring the network server, the interface as shown in Figure 34.1, “Add Hosts” is displayed.

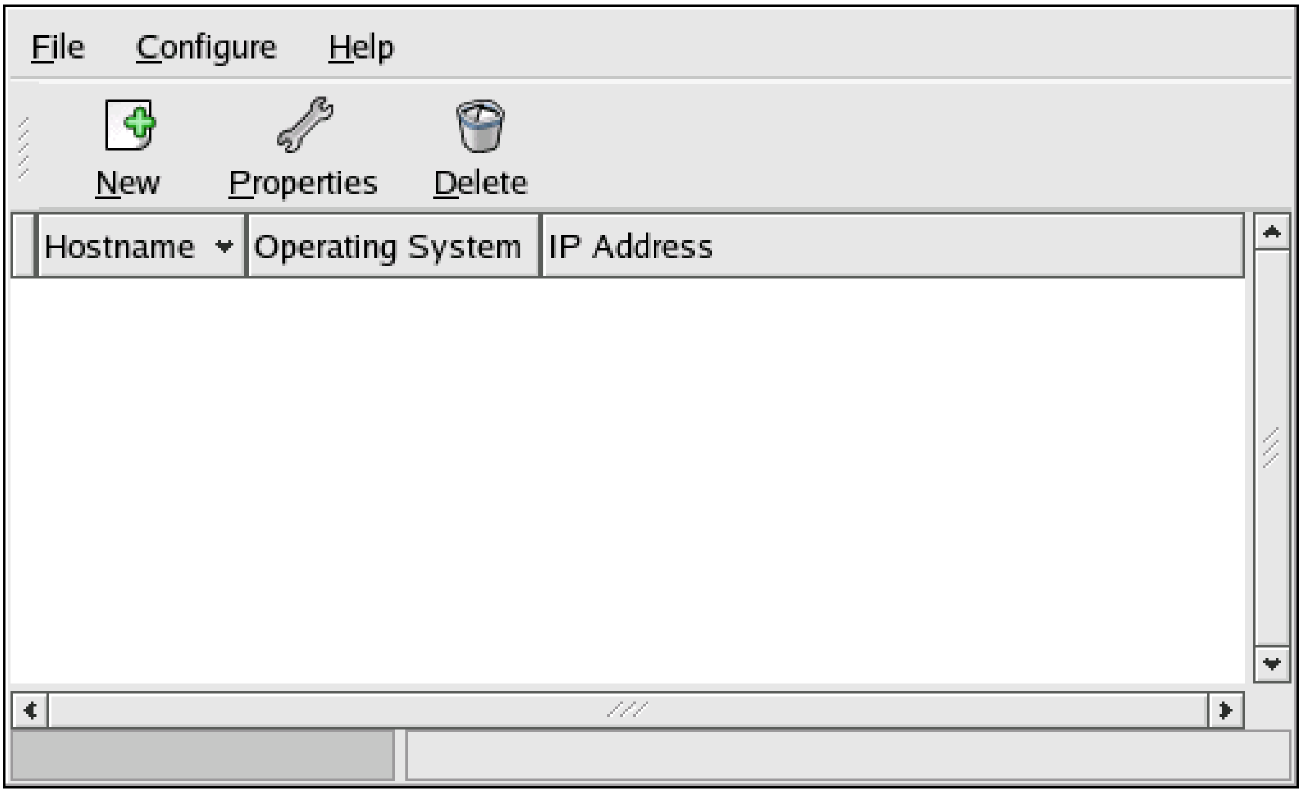

Figure 34.1. Add Hosts

The next step is to configure which hosts are allowed to connect to the PXE boot server. For the command line version of this step, refer to Section 34.3.1, “Command Line Configuration”.

To add hosts, click the **New** button.

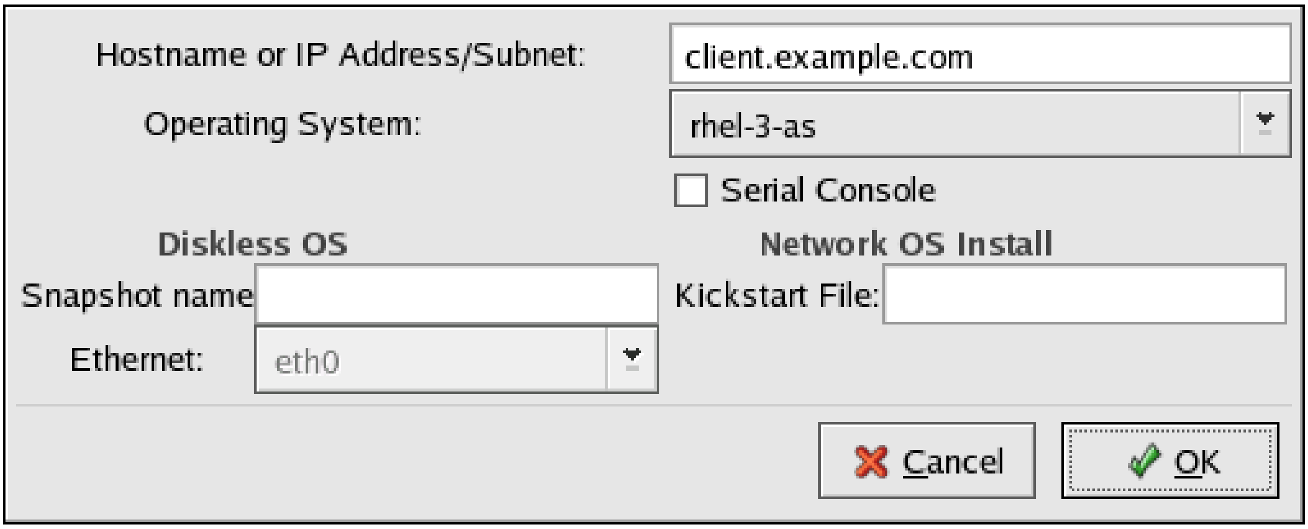

Figure 34.2. Add a Host

Enter the following information:

- **`Hostname or IP Address/Subnet`** — The IP address, fully qualified hostname, or a subnet of systems that should be allowed to connect to the PXE server for installations.

> **Multiple IP addresses not accepted**
>
> Only enter a single IP address. **anaconda** will not use multiple addresses.

- **`Operating System`** — The operating system identifier to install on this client. The list is populated from the network install instances created from the **`Network Installation Dialog`**.
- **`Serial Console`** — This option allows use of a serial console.
- **`Kickstart File`** — The location of a kickstart file to use, such as **`http://server.example.com/kickstart/ks.cfg`**. This file can be created with the **Kickstart Configurator**. Refer to Chapter 32, *Kickstart Configurator* for details.

Ignore the **`Snapshot name`** and **`Ethernet`** options. They are only used for diskless environments.

34.3.1. Command Line Configuration

If the network server is not running X, the **`pxeboot`** utility, a part of the **`system-config-netboot`** package, can be used to add hosts which are allowed to connect to the PXE server:

```
pxeboot -a -K <kickstart> -O <os-identifier> -r <value> <host>
```

The following list explains the options:

- **`-a`** — Specifies that a host is to be added.
- **`-K`** *<kickstart>* — The location of the kickstart file, if available.
- **`-O`** *<os-identifier>* — Specifies the operating system identifier as defined in Section 34.2, “PXE Boot Configuration”.
- **`-r`** *<value>* — Specifies the ram disk size.
- *<host>* — Specifies the IP address or hostname of the host to add.

For more information on command line options available for the **`pxeboot`** command, refer to the **`pxeboot`** man page.

34.4. TFTPD

34.4.1. Starting the `tftp` Server

On the DHCP server, verify that the **`tftp-server`** package is installed with the command **`rpm -q tftp-server`**. If it is not installed, install it via Red Hat Network or the Red Hat Enterprise Linux CD-ROMs.

> **Note**
>
> For more information on installing RPM packages, refer to the Package Management Section of the Red Hat Enterprise Linux Deployment Guide.

tftp is an xinetd-based service; start it with the following commands:

```
/sbin/chkconfig --level 345 xinetd on
/sbin/chkconfig --level 345 tftp on
```

These commands configure the **tftp** and **xinetd** services to immediately turn on and also configure them to start at boot time in runlevels 3, 4, and 5.

34.5. Configuring the DHCP Server

If a DHCP server does not already exist on the network, configure one. Refer to the Red Hat Enterprise Linux Deployment Guide for details. Make sure the configuration file contains the following so that PXE booting is enabled for systems which support it:

```
allow booting;
allow bootp;
class "pxeclients" {
    match if substring(option vendor-class-identifier, 0, 9) = "PXEClient";
    next-server <server-ip>;
    filename "linux-install/pxelinux.0"; }
```

where the next-server *<server-ip>* should be replaced with the IP address of the **tftp** server.

34.6. Adding a Custom Boot Message

Optionally, modify **/tftpboot/linux-install/msgs/boot.msg** to use a custom boot message.

34.7. Performing the PXE Installation

For instructions on how to configure the network interface card with PXE support to boot from the network, consult the documentation for the NIC. It varies slightly per card.

Revision History

Note that revision numbers relate to the edition of this manual, not to version numbers of Red Hat Enterprise Linux.

Revision 3.1-46	**Thu Sep 11 2014**	**Petr Bokoč**

Red Hat Enterprise Linux 5.11 GA release of the Installation Guide

Revision 3.1-45	**Mon Jun 30 2014**	**Petr Bokoč**

Red Hat Enterprise Linux 5.11 Beta release of the Installation Guide

Revision 3.1-44	**Mon Jun 30 2014**	**Petr Bokoč**

Updated the description of the **`network`** Kickstart command - BZ#1023106

Added an admonition about migration of root file system from singlepath to multipath storage being unsupported - BZ#1005029

Revision 3.1-43	**Thu Jul 11 2013**	**Petr Bokoč**

Red Hat Enterprise Linux 5.10 GA release of the Installation Guide

Revision 3.1-42	**Thu Jul 11 2013**	**Petr Bokoč**

Red Hat Enterprise Linux 5.10 Beta release of the Installation Guide

Revision 3.1-39	**Tue Apr 23 2013**	**Jack Reed**

Added author attributions to start of guide

Revision 3.1-38	**Tue Apr 23 2013**	**Jack Reed**

Clarified purpose of autostep kickstart argument - BZ#667560

Clarified wording of --trust option in firewall kickstart argument - BZ#813385

Corrected syntax for cdrom installation method parameter - BZ#855371

Noted that read-only devices must be specified with ignoredisk argument in kickstart installations - BZ#870450

Revision 3.1-37	**Mon Jan 7 2013**	**Jack Reed**

Clerical update and final build for 5.9 GA

Revision 3.1-36	**Mon Jan 7 2013**	**Jack Reed**

Version for 5.9 GA release

Revision 3.1-35	**Wed Oct 31 2012**	**Jack Reed**

Removing admonition titles in subscription chapter

Revision 3.1-34	**Wed Oct 31 2012**	**Jack Reed**

Updating registration and subscription chapter - BZ#871507

Revision 3.1-33	**Wed Sep 12 2012**	**Jack Reed**

Rebuilding for 5.9 Beta

Revision 3.1-32	**Thu Sep 06 2012**	**Jack Reed**

Beta Release

Revision 3.1-30	**Tue Aug 14 2012**	**Jack Reed**

Added size limit for ext3 - BZ#843305

Revision 3.1-26	Fri Feb 02 2012	Jack Reed

Updated FTP/HTTP installation prep procedure - BZ#240455
Corrected typos - BZ#766441, BZ#676559

Revision 3.1-24	Fri Nov 18 2011	Jack Reed

Explained values for **pv.<*id*>** in part and volgroup kickstart options - BZ#601834
Corrected current version number for display in the book - BZ#748681

Revision 3.1-22	Wed Nov 9 2011	Jack Reed

Updated clarification about packages during upgrade - BZ#748681
Improved screen output formatting in section 34.5 - BZ#752273

Revision 3.1-21	Fri Nov 4 2011	Jack Reed

Added home partition recommendation — BZ#747164
Clarified that third-party packages may not function properly after upgrade — BZ#748681
Added **--label=** directive to **part** or **partition** kickstart option — BZ#749693

Revision 3.1-20	Thu Oct 13 2011	Jack Reed

Added admonition regarding boot USB names — BZ#745665

Revision 3.1-19	Thu Sep 29 2011	Jack Reed

Added admonition to Adding PXE Hosts section — BZ#660316
Fixed dead links in chapter 25 — BZ#604871

Revision 3.1-18	Fri Sep 23 2011	Jack Reed

Added section on deregistering from entitlement platforms — BZ#604871

Revision 3.1-17	Mon Sep 19 2011	Jack Reed

Noted only one IP address accepted in PXE installation — BZ#660316
Edited and removed figures from System z text mode installation section — BZ#718948

Revision 3.1-16	Thu Jul 21 2011	Rüdiger Landmann

Fix ordering of kickstart options — BZ#701378
Describe default behavior of --ethtool option — BZ#674475
Fix typo — BZ#317221
Correction about RAID setup with IPR card on PPC — BZ#683620
Document supplied redhat.exe REXX script — BZ#559969

Revision 3.1-15	Fri Jun 10 2011	Rüdiger Landmann

Document noeject option — BZ#668995

Revision 3.1-14	Mon Apr 11 2011	Rüdiger Landmann

Typo in script — RT3#104480

Revision 3.1-13	Thu Jan 6 2011	Rüdiger Landmann

Correct instructions to create USB bootable media — BZ#317221

Revision 3.1-12	Wed Jan 5 2011	Rüdiger Landmann

Correct pxeos example — BZ#243098

Revision 3.1-11	Wed Jan 5 2011	Rüdiger Landmann

Clarify pxeos details — BZ#243098

Revision 3.1-10	Wed Jan 5 2011	Rüdiger Landmann

Document using clearpart to ensure that DASDs are formatted during installation — BZ#606048
Expand documentation of multipath kickstart command — BZ#629834

Revision 3.1-9	Tue Dec 21 2010	Rüdiger Landmann

Update Figure 4.14. Configure ISCSI Parameters — BZ#560869
Describe --log option for Kickstart %post scriptlets — BZ#568873
Add list of package groups — BZ#577326
Correct description of Kickstart completion options — BZ#580692
Document ignordisk --only-use option — BZ#591745
Document --fsoptions for part kickstart command — BZ#600298
Fix assorted typos — BZ#653251

Revision 3.1-5	Thu Oct 7 2010	Rüdiger Landmann

Bump release number to build

Revision 3.1-2	Mon Apr 19 2010	Rüdiger Landmann

Bump release number to build

Revision 3.1-1	Mon Apr 19 2010	Rüdiger Landmann

Fix RAID levels in Kickstart documentation -- BZ#561979

Revision 3.1-0	Tue Apr 6 2010	Rüdiger Landmann

Bump version for release

Revision 3.0-23	Tue Apr 6 2010	Rüdiger Landmann

Fix some broken links -- BZ#561247
Update menu name from "Package Updater" to "Software Updater" -- BZ#563775
Note that Kickstart Configurator is not installed by default -- BZ#564257
Document "specify installation key" option for Kickstart Configurator -- BZ#564262
Update version number throughout doc -- BZ#564265
Added descriptions of when to use the lcs and qeth NETTYPE options for System z parm files -- BZ#576787
Update version number throughout doc -- BZ#564265
Note that /var cannot be a separate partition on network storage -- BZ#577695
Document the --passalgo option for the auth kickstart command -- BZ#578664
Document installation on multipath devices -- BZ#522856
Update Edit Interfaces screen -- BZ#560878

Revision 3.0-22	Wed Mar 31 2010	Rüdiger Landmann

Document nostorage kickstart option -- BZ#526630
Correct order of quotemarks in boot parameters -- BZ#526631
Document Enable network Interface screen and reverse CHAP parameters for iSCSI -- BZ#560869
Document encryption option when adding a partition -- BZ#560875

Revision 3.0-21	Tue Jan 12 2010	Rüdiger Landmann

Everything installations not supported -- BZ#549898

Revision 3.0-20	Tue Jan 12 2010	Rüdiger Landmann

Explain RAID issues BZ#486735

Revision 3.0-19	**Mon Jan 11 2010**	**Rüdiger Landmann**

Fix some line breaks per BZ#486735

Revision 3.0-18	**Mon Jan 11 2010**	**Rüdiger Landmann**

Remove **`nokill`** as a kickstart command BZ#513693

Revision 3.0-17	**Mon Jan 11 2010**	**Rüdiger Landmann**

Link to Red Hat Knowledgebase article on RAID 1 failures BZ#486735

Revision 3.0-16	**Mon Jan 11 2010**	**Rüdiger Landmann**

Document --hvargs option for **`bootloader`** kickstart command BZ#553321

Revision 3.0-15	**Fri Jan 8 2010**	**Rüdiger Landmann**

Update anaconda homepage BZ#486735
Remove exclamation mark from echo command BZ#486735
split /sbin/shutdown and /sbin/chkconfig examples onto separate lines BZ#486735

Revision 3.0-14	**Tue Dec 22 2009**	**Rüdiger Landmann**

Include note about boot time logging not available BZ#549608
Include note about global passphrases not supported BZ#549609

Revision 3.0-13	**Mon Dec 21 2009**	**Rüdiger Landmann**

Include reverse CHAP parameters for iscsi kickstart command BZ#525139

Revision 3.0-12	**Thu Dec 17 2009**	**Rüdiger Landmann**

Document iscsi and iscsiname kickstart command BZ#525139

Revision 3.0-11	**Wed Dec 16 2009**	**Rüdiger Landmann**

Note about improved swap performance per BZ#503878

Revision 3.0-10	**Fri Dec 11 2009**	**Rüdiger Landmann**

Update swap recommendations per BZ#542865

Revision 3.0-9	**Fri Dec 11 2009**	**Rüdiger Landmann**

correct missing entity
add standard Preface

Revision 3.0-8	**Fri Dec 11 2009**	**Rüdiger Landmann**

revised Legal Notice, removed restrictions

Revision 3.0-1	**Tue Aug 04 2009**	**Rüdiger Landmann, Jon Masters**

Numerous bug fixes, new chapters on driver updates

Revision 2.0-1	**Mon Jan 05 2009**	**Don Domingo**

revised Legal Notice, removed restrictions

Revision 1.0-1	**Fri Oct 03 2008**	**Don Domingo**

migrated to new automated build system

Index

Symbols